



TRƯỜNG ĐẠI HỌC FPT

MINISTRY OF EDUCATION AND TRAINING

FPT UNIVERSITY

Capstone Project Document

[Clinic Information System]

SWP490_G14

Luu Cao Duy – SE04194

Vu Van Quan – SE05408

Group Members Hoang Viet – HE 130292

Bui Huu Minh Tri – HE130497

Nguyen Hoang Linh – HE130961

Supervisor Ms. Tran Thu Thuy

Capstone Project Code CIS

- Hanoi, April 2022 -

ACKNOWLEDGEMENT.....	7
DEFINITION AND ACRONYMS	7
I. PROJECT INTRODUCTION	9
1. OVERVIEW	9
1.1 Project Information	9
1.2 Project Team.....	9
2. PRODUCT BACKGROUND	10
3. EXISTING SYSTEMS	10
3.1 FMP website	10
3.2 ISOFH CARE.....	12
4. BUSINESS OPPORTUNITY	12
5. SOFTWARE PRODUCT VISION	13
6. PROJECT SCOPE & LIMITATIONS.....	14
6.1 Major Features	14
6.2 Limitations & Exclusions	15
II. PROJECT MANAGEMENT PLAN	16
1. OVERVIEW	16
1.1 Scope & Estimation.....	16
1.2 Project Objectives	22
1.3 Project Risks	23
2. MANAGEMENT APPROACH	24
2.1 Project Process.....	24
2.2 Quality Management	25
2.2.1 Defect Prevention:.....	25
2.2.2 Reviewing:.....	25
2.2.3 Unit Testing:.....	26
2.2.4 Integration Testing:.....	26
2.2.5 System Testing:.....	26
2.3 Training Plan	26
3. PROJECT DELIVERABLES.....	27
4. RESPONSIBILITY ASSIGNMENTS	29
4.1 Team & Structures:	29
4.2 Roles & Responsibilities:	29
5. PROJECT COMMUNICATIONS	31
5.1 Communication Plan.....	31
5.2 External Interface	32
5.2.1 FU Contacts	32
5.2.2 Customer Contacts	32
6. CONFIGURATION MANAGEMENT.....	32
6.1 Document Management.....	32
6.2 Source Code Management.....	32
6.3 Tools & Infrastructures	33
III. SOFTWARE REQUIREMENT SPECIFICATION.....	34
1. PRODUCT OVERVIEW.....	34
2. USER REQUIREMENTS.....	35
2.1 Actors	35
2.2 Use Cases.....	36
2.2.1 Diagrams	36
2.2.2 Use Case List.....	38
3. FUNCTIONAL REQUIREMENTS	42
3.1 System Functional Overview	42

3.1.1 Screens Flow	42
3.1.2 Screen Descriptions.....	44
3.1.3 Screen Authorization	46
3.1.4 Non-Screen Functions	47
3.1.5 Entity Relationship Diagram.....	48
3.2 Homepage	50
3.3 Authentication.....	51
3.3.1 Login	51
3.4 Manage Appointment	52
3.4.1. View List Appointment.....	52
3.4.2 Appointment Detail	53
3.4.3 Create New Appointment.....	54
3.4.4 Accept Appointment.....	55
3.4.5 Delete Appointment	56
3.5 Manage Doctor Information.....	57
3.5.1 View List Doctor.....	57
3.5.2 Doctor Information Detail	58
3.5.3 Create New Doctor Information	59
3.5.4 Update Doctor Information	60
3.5.5 Delete Doctor Information.....	61
3.6 Manage Doctor Schedule.....	62
3.6.1 View Schedule.....	62
3.6.2 Update Doctor Schedule	63
3.7 Manage Patient Information	64
3.7.1 View List Patient	64
3.7.2 Patient Information Detail.....	65
3.7.3 Create New Patient Information.....	66
3.7.4 Update Patient Information	67
3.7.5 Delete Patient Information.....	68
3.8 Manage Medical Record	69
3.8.1 Medical Record Detail	69
3.8.2 Create New Medical Record	70
3.8.3 Print prescriptions	71
3.9 Manage Clinic.....	72
3.9.1 Manage patient list in clinic	72
3.10 Manage Treatment.....	73
3.10.1 View List Treatment.....	73
3.11 Manage Category	74
3.11.1 View List Categories	74
3.11.2 Edit Categories	75
3.11.3 Delete Categories	76
3.11.4 Create New Categories.....	77
4. NON-FUNCTIONAL REQUIREMENTS.....	78
4.1 External Interfaces.....	78
4.1.1 User Interfaces.....	78
4.1.2 Software Interfaces	78
4.1.3 Hardware Interfaces.....	78
4.1.4 Communications Interfaces.....	78
4.2 Quality Attributes	79
4.2.1 Usability.....	79
4.2.2 Reliability.....	79
4.2.3 Performance	79
4.2.4 Dependability	79
4.2.4.1 Security	79
4.2.4.2 Safety	79
4.2.5 Supportability	79
4.2.6 Design Constraints.....	80

4.2.7 Support Documents	80
5. REQUIREMENT APPENDIX	80
5.1 Business Rules.....	80
5.2 Common Requirements	81
5.3 Application Messages List.....	81
5.4 Other Requirements.....	81
IV. SOFTWARE DESIGN DESCRIPTION.....	82
1. SYSTEM DESIGN	82
1.1 System Architecture.....	82
1.1.1 Overall Architecture.....	82
1.1.1.1 System Architectural Diagram	82
1.1.1.2 Component Explanation	82
1.1.1.2.1 Front-end of Web Application.....	82
1.1.1.2.2 Back-end of Web Application.....	82
1.1.1.3 Technology and Framework Explanation.....	82
1.1.1.3.1 Bootstrap 3	82
1.1.1.3.2 Spring Boot	83
1.1.1.3.3 MySQL.....	83
1.1.1.3.4 IntelliJ IDEA	84
1.1.2 F-CIS Architecture	85
1.2 Package Diagram	87
1.2.1 F- CIS Front-end	87
1.2.1.1 Diagram	87
1.2.1.2 Package Descriptions.....	87
1.2.2 F- CIS Back-end	88
1.2.2.1 Diagram	88
1.2.2.2 Package Descriptions.....	89
2. DATABASE DESIGN.....	89
2.1. Class Specifications	89
2.1.1 CIS Back-end	89
2.1.1.1 Controller.....	89
2.1.1.1.1 UserRestController	89
2.1.1.1.2 CatalogController	90
2.1.1.1.3 AppController (Web)	91
2.1.1.2 Service.....	95
2.1.1.2.1 UserService	95
2.1.1.2.2 ServiceService.....	96
2.1.1.2.3 MedicineService	96
2.1.1.2.4 DiseaseListICD10Service	96
2.1.1.2.5 MasterDataService	97
2.1.1.2.6 MasterDataConfigService	97
2.1.1.2.7 SpecializedService	97
2.1.1.2.8 PrescriptionOfPatientService	98
2.1.1.2.9 PatientService	98
2.1.1.2.10 PatientService	99
2.1.1.2.11 GroupMasterService	99
2.1.1.2.12 ExaminationRoomService	99
2.1.1.2.13 CMSRegimentService	100
2.1.1.2.14 AdmissionRecordService	100
2.1.2 CIS Front-end	101
2.1.2.1 Pages	101
2.1.2.1.1 Admin	101
2.1.2.1.2 Receptionist	102
2.1.2.1.3 Doctor	103
2.1.2.1.4 Patient	104
2.1.2.1.5 Common.....	104
2.1.2.2 Library.....	104

2.1.2.2.1 Common.....	104
2.1.2.3 Context	105
2.1.2.3.1 Authenticator.....	105
2.1.2.4 Components.....	106
2.1.2.4.1 UI	106
2.1.2.4.2 Layout	106
2.2. Database Design.....	108
2.2.1 App_User	108
2.2.2 Admission_record	109
2.2.3 Examination_room	110
2.2.4 Generic_master	111
2.2.5 Group_master	111
2.2.6 Medicine_route	112
2.2.7 Medicine_master.....	112
2.2.8 Medicine_specialized.....	112
2.2.9 Patient	113
2.2.10 Patient_diagnosis	115
2.2.11 Prescription_of_patient.....	115
2.2.12 Disease_list_icd10	116
2.2.13 Regimen.....	117
2.2.14 Regimen_medicine	117
2.2.15 Regimen_service.....	118
2.2.16 Regimen_specialized	118
2.2.17 Role_type.....	119
2.2.18 Room	119
2.2.19 Route_master	119
2.2.20 Service_master	120
2.2.21 Service_of_patient.....	120
2.2.22. Service_price_history.....	121
2.2.23. Service_specialized	121
3. DETAILED DESIGN	122
3.1 Login.....	122
3.1.1 Class Diagram.....	122
3.1.2 Sequence Diagram Login	122
3.2 Admin Management (Create)	123
3.2.1 Class Diagram CreateDoctorAccount.....	123
3.2.2 Sequence Diagram CreateDoctorAccount	123
3.3 Admin Management (Update)	124
3.3.1 Class Diagram UpdateDoctorAccount	124
3.3.2 Sequence Diagram UpdateDoctorAccount	124
3.3 Admin Management (Delete)	125
3.3.1 Class Diagram Delete DoctorAccount.....	125
3.3.2 Sequence Diagram Delete DoctorAccount	125
3.4 Threading Patients	126
3.4.1 Class Diagram Threading Patients.....	126
3.4.2 Sequence Diagram Threading Patients.....	126
3.5 Make Appointment.....	127
3.5.1 Class Diagram MakeAnAppointment	127
3.5.2 Sequence Diagram MakeAnAppointment	127
3.6 Doctor Examination	128
3.6.1 Class Diagram Doctor Examination	128
3.6.2 Sequence Diagram Doctor Examination	128
3.7 Medical Instruction	129
3.7.1 Class Diagram Medical Instruction	129
3.7.2 Sequence Diagram Medical Instruction	129
3.8 Print Prescription	130
3.8.1 Class Diagram Print Prescription	130
3.8.2 Sequence Diagram Print Prescription	130

3.9 Payment	131
3.9.1 Class Diagram Payment.....	131
3.9.2 Sequence Diagram Payment.....	131
V. SOFTWARE TESTING DOCUMENTATION	132
1. SCOPE OF TESTING	132
1.1 Test Model.....	132
2. TEST STRATEGY	133
2.1 Testing Types.....	133
2.1.1. Function Testing	133
2.1.2. API Testing.....	134
2.1.3. UI/UX Testing	134
2.2 Test Levels	134
2.2.1. Unit Testing	134
2.2.2. Integration Testing	135
2.2.3. System Testing.....	136
2.3 Supporting Tools.....	137
3. TEST PLAN	137
3.1 Human Resources	137
3.2 Test Environment.....	137
3.3 Test Milestones.....	138
4. TEST CASES	139
4.1 Unit Testing	139
4.2 Integration Testing & System Testing	140
5. TEST REPORTS.....	142
5.1 Unit Test Report.....	142
5.2 Integration Test Report.....	142
5.3 System Test Report	143
5.4 Total LOC	143
VI. RELEASE PACKAGE & USER GUIDES.....	144
1. DELIVERABLE PACKAGE	144
1.1 Source codes & documents	144
1.2 Known Issues, Limitations & Restrictions	145
2. INSTALLATION GUIDES	145
2.1 System Requirements	145
2.2 Installation Instruction.....	145
3. USER MANUAL	146
3.1 System requirements	146
3.2 Application Usage.....	146
3.2.1 Guest	146
3.2.2 Doctor.....	147
3.2.3 Patient.....	152
3.2.4 Admin	153
3.2.5 Receptionist.....	159
3.3 Troubleshooting.....	162
VII. APPENDIX	162

Acknowledgement

This project would have not been completed without the support of many people. So, we want to thank all of them.

First of all, we would like to express our deep gratitude to Ms. Tran Thu Thuy, our supervisor, for his patient guidance, enthusiastic encouragement and useful critiques of this project. Under the instructions of Ms. Thuy, the project team has always felt comfortable and confident. Ms. Thuy shared with us the philosophy of keeping the right attitude towards problems. We believe that was the most important factor that led to the success of this project.

Besides, we would also like to extend our thanks to the instructors at FPT University for all the classes. We hope you will find this project as a reflection of the knowledge and experiences you have given us during this period of studying years.

Finally, we wish to thank our parents for their support and encouragement throughout our study.

Thank all of you.

Definition and Acronyms

Acronym	Definition
API	Application Program Interface
BA	Business Analysis
BR	Business Rule
CRUD	Create, Read, Update, Delete
DB	Database
DBMS	Database Management System
EC2	Elastic Compute Cloud
ERD	Entity Relationship Diagram

FK	Foreign Key
IDE	Integrated Development Environment
IT	Information Technology
JS	Javascript
PK	Primary Key
PM	Project Manager
QA	Quality Assurance
RDS	Relational Database Service
S3	Simple Storage Service
SRS	Software Requirement Specification
UAT	User Acceptance Test
UC	Use Case
UI	User Interface
UML	Unified Modelling Language
UX	User Experience
WBS	Work Breakdown Structure

I. Project Introduction

1. Overview

1.1 Project Information

- Project name: **Clinic Information System**
- Project code: **CIS**
- Group name: **SWP490_G14**
- Software type: **Website**

1.2 Project Team

a. Supervisor

Full Name	Role	Email	Mobile
Tran Thu Thuy	Lecturer	thuytt108@fe.edu.vn	0904380878

b. Team Members

Full Name	Role	Email	Mobile
Luu Cao Duy	Leader	duylcse04194@fpt.edu.vn	0777414800
Nguyen Hoang Linh	Member	linhnhhe130961@fpt.edu.vn	0868130599
Vu Van Quan	Member	quanvvse05408@fpt.edu.vn	0385617419
Bui Huu Minh Tri	Member	tribhmhe130497@fpt.edu.vn	0352524027
Hoang Viet	Member	viethhe130292@fpt.edu.vn	0395558546

2. Product Background

Today, medical staff at clinics face many difficulties in patient management and distribution, especially in the context of an epidemic.

Most of this difficulty comes from the fact that the clinic does not have a specific patient stream and management system, medical staff have to perform manual allocation, leading to difficulty in control, coordination and loss of time to handle this task.

The absence of a management system leads to the loss of patient information and time to collect patient information, making the waiting room always overloaded. So, systematization of information for clinics is necessary in the current situation.

When there are many patients coming for medical examination and treatment, doctors and medical staff need to control the number of patients accurately and effectively classify them. Clinic management system helps them to systematize the medical examination and treatment process of patients with doctors.

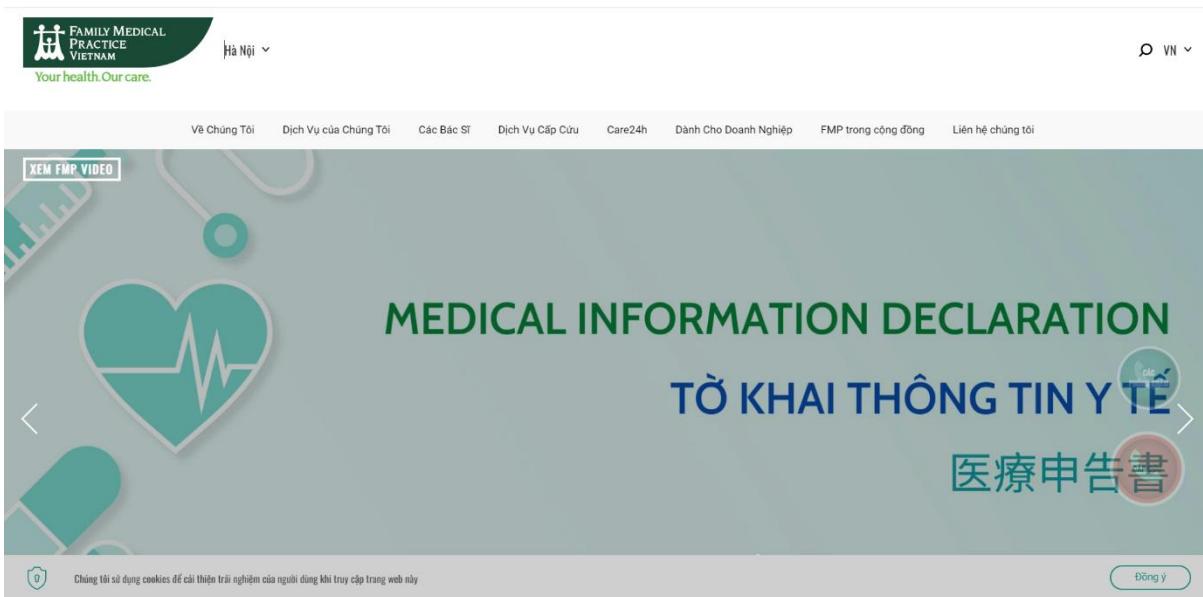
We provide a clear system of tasks and connections between medical staff - patients - doctors in the clinic. This helps medical staff to grasp the specific schedule of doctors. From there, medical staff and doctors can dispose, co-ordinate, triage quickly and appropriately. Help ensure time efficiency and quality in the clinic. To minimise bulky and reduce crowds, to help clinic operate smoothly, maximum working speed and efficiency.

Therefore, the CIS was created as to solve a number of problems in clinics, contributing to the overall development of the medical industry.

3. Existing Systems

3.1 FMP website

FMP as the first foreign-owned and operated primary health care provider in Vietnam, Family Medical Practice has consistently remained at the forefront of international-standard medicine since our opening in 1995. Friendly interactive interface that is easy to manipulate with all objects.



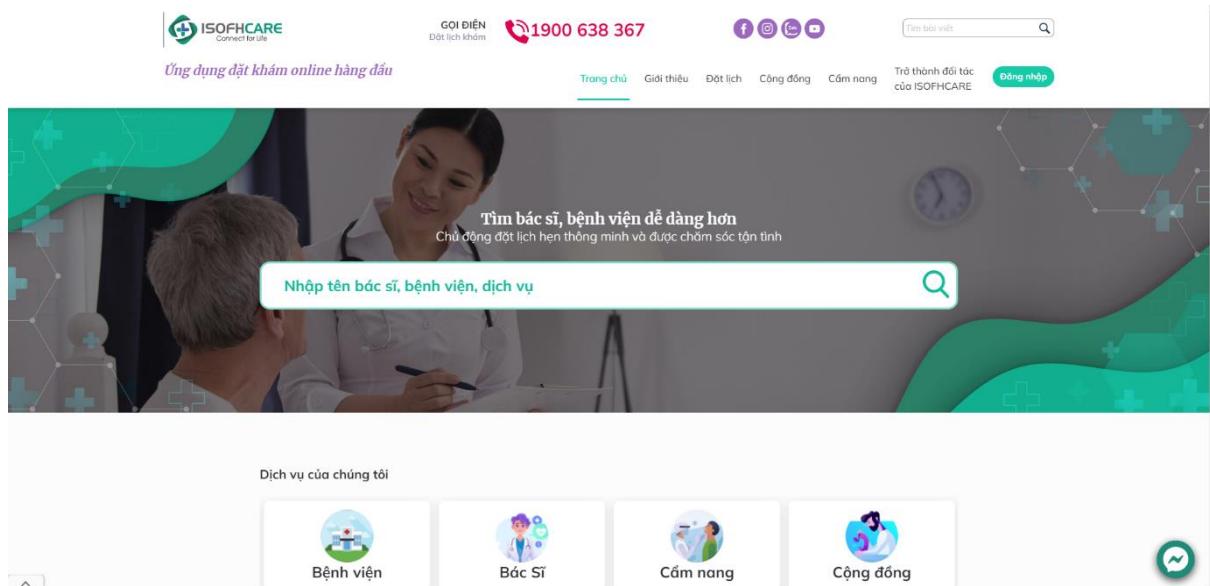
Summary of the advantages and disadvantages of FMP

Advantage	Disadvantage
<ul style="list-style-type: none">Have links from clinics.Have an introduction to health checkup packages.	<ul style="list-style-type: none">Can't make an appointment.Just a page that advertises services.

Table 1 - Advantages and disadvantages of FMP

3.2 ISOHCARE

The system is linked with the clinic, hospital, doctor, providing and processing the requests of patients who need medical examination and treatment and sent to affiliated units. This application, simple, friendly, easy to use for first time users with many health monitoring support features.



Summary of the advantages and disadvantages of ISOHCARE

Advantage	Disadvantage
<ul style="list-style-type: none">Schedule an appointment with a specific date and time, patient can choose doctor, hospital, according to disease.Has developed on two platforms: Application and Web, convenient for many audiences.	<ul style="list-style-type: none">Just a partner providing associated services, collecting information for clinics, hospital. Is not a clinic management system, without a specific location.Unable to store patient information and medical files on the system.

Table 2 - Advantages and disadvantages of ISOHCARE

4. Business Opportunity

In an era of competition, any business providing professional services makes customers trust and choose the service. For employees, they will be able to work in a professional digitized environment to help reduce the amount of work. The system will save time for patients and staff, help

the clinic access patient information when needed quickly and accurately, as well as management costs for the clinic.

To manage the clinic and work in the clinic to be controlled more effectively, a clinic management system was born to do just that. With this management system, medical staff can control the doctor's work and schedule easily, saving time and effort compared to traditional paper management. From the can better control and triage the patient. When a patient comes for medical examination and treatment, medical staff can base on the medical record and schedule, triage the patient to the doctor's clinic quickly and track the patient's case-record. The system will save the patient's information and the diagnostic information of the doctor receiving medical examination and treatment.

The system helps to improve the progress of work in the clinic without delay. It takes less time for the medical staff than for the medical staff to control the doctor's schedule and the patient's case-record, management and triage with manual notes. The clinic management system is suitable for all types of clinics, large clinics to small clinics, polyclinics and specialties.

5. Software Product Vision

Clinic management software is a digital tool that allows you to receive patient information, create and track medical records, organize and review your doctor's work schedule. Moreover, it provides a system to automate patient-doctor visits with the goal of improving clinic quality, business results, and clinic performance. It helps them plan work, manage the time of medical staff and doctors as well as improve efficiency.

A good management system will automate processes, eliminate paperwork and cut down on tedious tasks in order to allow doctors to focus on professional work. We want to create a system to improve the quality of clinics in particular as well as the country's healthcare in general. This system not only assists patients in having a better experience but also helps medical staff and doctors to do their job properly without wasting time by tedious paperwork.

Develop a commercial website into an app to sell to clinics and hospitals. From there, helping the country's health system to develop more starting from small clinics. A clear medical information system that helps hospitals in the process of merging or decoupling easily.

6. Project Scope & Limitations

6.1 Major Features

- There are 4 types of users in CIS System:

Type of User	Description
Admin	People who manage the CIS System.
Receptionist	People who confirm appointment, schedule, receiving and threading patients.
Patient	People who use the system to make an appointment.
Doctor	People who use the system to make medical instructions, examine, treatment in medical record.

- CIS System will provide the following functions to the above users:

FE	Description
01	Patients can book an appointment with a doctor.
02	Receptionist can book an appointment for new patient at clinic.
03	Receptionist manages patient information.
04	Receptionist threading the patient.
05	Doctors can view and edit information examine, treatment in medical record of patient have code list disease by the standards of the Ministry of Health. Code: ICD10.

06	Doctors make medical instructions.
07	Medical record. (The data about the results of examination and diagnosis will be saved in the patient's file => Making accurate diagnosis conclusions from the doctor and generating ICD10 codes. => Create prescriptions for patients => Print)
08	Print invoices and prescriptions.
09	Admin manage categories.

Table 3 - Major Features

6.2 Limitations & Exclusions

- Connect with pharmacy.
- Print barcodes containing patient information.
- Connect with para-clinical diagnosis.
- Patients can check the information of the doctors working at the clinic.
- Patients can see the doctor's work schedule for the week.
- Patients can view their medical record.

II. Project Management Plan

1. Overview

1.1 Scope & Estimation

WBS ID	WBS Item	Complexity	Est. Effort (man-days)
1	Project Initiating		<u>18</u>
1.1	Feasibility Study	Complex	5
1.2	List up requirements	Complex	6
1.3	Create report 1 (Project Introduction)	Medium	5
1.4	Kick-off meeting	Simple	1
1.5	Finish project initiating	Simple	1
2	Project Planning		<u>34</u>
2.1	Team meeting	Medium	22
2.2	Meet instructor	Medium	0.5
2.3	Training technical	Complex	15
2.4	Create report 2 (Project Management Plan)	Complex	6
2.5	Create report 5 (Test Plan)	Complex	6
2.6	Finish Project planning	Simple	0.5
3	Project Executing		<u>326</u>

3.1	Requirement & Design		38
3.1.1	Define Scope	Complex	7
3.1.2	Analyze requirements	Complex	7
3.1.3	Create report 3 (Software Requirement Specification)	Complex	7
3.1.4	Create technical prototype	Medium	10
3.1.5	Create report 4 (Software Design Specification)	Medium	7
3.2	Code & Testing		288
3.2.1	Integration 1		70
3.2.1.1	Authentication		21
3.2.1.1.1	Login	Simple	3
3.2.1.1.2	Forget password	Complex	8
3.2.1.1.3	Sign in	Medium	5
3.2.1.1.4	Change account information	Medium	5
3.2.1.2	Make appointment		8
3.2.1.2.1	Create appointment form	Complex	8
3.2.1.3	Reception		31
3.2.1.3.1	Appointment List Screen	Medium	7
3.2.1.3.2	Appointment Detail Screen	Medium	7

3.2.1.3.3	Reception Screen	Medium	5
3.2.1.3.4	Patient Register Screen	Medium	7
3.2.1.3.5	Patient Appointment Screen	Medium	5
3.2.1.4	Testing (Integration test & unit test)		10
3.2.1.4.1	Create test plan	Simple	1
3.2.1.4.2	Create unit test case	Medium	3
3.2.1.4.3	Create test case	Medium	3
3.2.1.4.4	Test integration 1	Simple	1
3.2.1.4.5	Verify test integration 1	Simple	1
3.2.1.4.6	Fix bug integration 1	Medium	1
3.2.2	Integration 2		123
3.2.2.1	Manage Patients Profile		21
3.2.2.2.1	List patient	Medium	8
3.2.2.2.2	Add and update patient	Medium	5
3.2.2.2.3	Patient information	Complex	8
3.2.2.2	Manage Doctor Information		28
3.2.2.2.1	List doctor	Complex	8
3.2.2.2.2	Add and update information doctor	Complex	8

3.2.2.2.3	Add and update schedule doctor	Medium	5
3.2.2.2.4	Doctor information	Medium	7
3.2.3.1	Manage Medical Record		44
3.2.3.1.1	Add and update medical record	Medium	8
3.2.3.1.2	Patient Health record history list	Medium	8
3.2.3.1.3	Patient Health detail record	Medium	8
3.2.3.1.4	Examination Screen	Medium	8
3.2.3.1.5	Create prescriptions, instructions, medical order	Complex	12
3.2.2.4	Testing (Integration test & unit test)		20
3.2.2.4.1	Create test plan	Simple	2
3.2.2.4.2	Create unit test case	Medium	4
3.2.2.4.3	Create test case integration 2	Medium	4
3.2.2.4.4	Test integration 2	Simple	3
3.2.2.4.5	Verify test integration 2	Simple	3
3.2.2.4.6	Fix bug integration 2	Medium	4
3.2.2.5	Document		10
3.2.2.5.1	Create report 4 (Software Design)	Medium	10
3.2.3	Integration 3		95

3.2.3.1	Print invoices and prescriptions		5
3.2.3.2.1	Print invoices	Simple	2
3.2.3.2.2	Print prescriptions	Simple	3
3.2.3.2	Patient threading		16
3.2.3.3.1	Show list patient	Complex	10
3.2.3.3.2	Show profile patient	Medium	6
3.2.3.4	Create clinical path		16
3.2.3.4.1	Show list clinical path	Medium	8
3.2.3.4.2	Add, update and delete clinical path	Medium	8
3.2.3.5	Testing (Integration test & unit test)		20
3.2.3.5.1	Create test plan	Simple	2
3.2.3.5.2	Create unit test case	Medium	4
3.2.3.5.3	Create test case	Medium	4
3.2.3.5.4	Test integration 3	Simple	3
3.2.3.5.5	Verify test integration 3	Simple	3
3.2.3.5.6	Fix bug integration 3	Medium	4
3.2.3.5	Manage categories		28
3.2.3.5.1	Category of technical services	Medium	7

3.2.3.5.2	Category of test codes	Medium	7
3.2.3.5.3	Category of diseases ICD 10	Medium	7
3.2.3.5.4	Category of medicine	Medium	7
3.2.3.6	Document		10
3.2.3.6.1	Create report 5 (Test Document)	Medium	10
4	Project Monitoring and Controlling		<u>20</u>
4.1	System Testing	Medium	8
4.2	Monitor and control scope	Medium	4
4.3	Monitor and control risk	Medium	4
4.4	Tracking progress	Medium	4
5	Project Closing		<u>22</u>
5.1	Create report 6 (Software User Guides)	Complex	6
5.2	Create report 7 (Final Project)	Complex	8
5.3	Team meeting	Simple	2
5.4	UAT	Complex	5
5.5	Close project	Medium	1
Total Estimated Effort (man-days)			420

1.2 Project Objectives

- Timeliness: The project must be finished before April 27, 2022.
- Allocated Effort (man-days): 420
- Defect Distribution:

#	Testing Stage	Test Coverage	No. of Defects	% of Defect	Notes
1	Reviewing	3.7%	22	10.3%	Technical leader reviews code of members before merging
2	Unit Test	50.6%	108	50.5%	Developer creates and tests
3	Integration Test	33.8%	58	27.1%	The tester creates and tests
4	System Test	11.9%	26	12.1%	The tester creates and tests
	Total	100%	214	100%	

1.3 Project Risks

No	Risk Description	Impact	Possibility	Response Plans
1	The member don't have enough ability to complete the tasks	High	Medium	<p>Training should be done before doing a task.</p> <p>Assign jobs in accordance with each member's working ability.</p>
2	Lack Management Skill	High	High	Study about the planning, monitoring and controlling process of project management.
3	Not having a clear scope	High	Medium	All members should discuss carefully at the beginning of each integration to define scope and requirement.
4	Members are lazy, not interested in doing job	High	Low	<p>Motivating the members by meeting, going out.</p> <p>Talk and ask about each person's situation.</p>
5	Member may sick and unable to work	Medium	Low	Ensure resource allocation is modified correctly to adapt to new requirement
6	Due to the influence of COVID-19, team cannot directly resolve issues together	Medium	High	Online meeting together to solve this issues.
7	Internet connection is down and team members cannot work online, submit work or merge code	Medium	Low	<p>All developer has to setup the isolated development environment and have an offline copy of the document</p> <p>Use alternatives to connect to the internet such as 4G, free wifi etc.</p>

8	System database are down	High	Medium	Avoid making heavy request or asking too many question Restart database service Test before possible data errors
9	Working progress is behind schedule	High	Low	Involve whole team members in estimation, perform daily progress meetings, overtime working to recovery schedule if needed. Prioritising feature: identify what must have, should have and nice to have.
10	Computer and laptop breakdown	High	Low	All developer has an online copy of the document and code. Try to borrow another device while waiting for repair.

2. Management Approach

2.1 Project Process

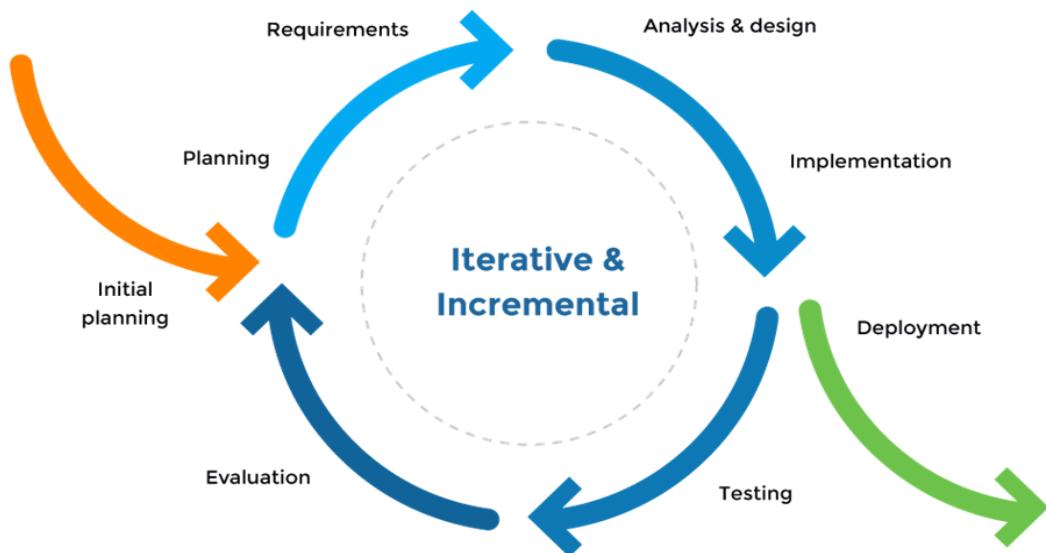


Figure 2.1: Iterative & Incremental Software Process Model

In all development models, CIS (Clinic Information System) projects shall use the Iterative and Incremental Development to software development. Iterative and Incremental Development is often used in big projects. The requirements are well defined, some details are corrected after each integration. With the method of subdividing each version, the big cycles are divided into small cycles corresponding to each copy. So, we can capture and control the development of the project easily and flexibly and easily manage. After studying the software development model carefully, we realise that my project is suitable for using the Iterative and Incremental Development model. That makes it easy for us to manage and develop the project by Iterative and Incremental Development model.

Also, this model has these useful advantages:

- Main function can be developed early and quickly.
- Easily manage risks that may arise.
- Splitting into small cycles after each integration makes easy testing and debugging.
- Customers can soon approach important functions.
- Customers can provide feedback to each product increment, thus avoiding surprises at the end of development.

2.2 Quality Management

2.2.1 Defect Prevention:

- Directly communicate with the person related to the defect, if the defect is detected early.
- Root cause analysis and defect notes, then take corrective action.
- Analyse and evaluate the extent of the defect, thereby giving a solution to the defect.
- So as not to affect the system, depends on how bad the defect is. Give a clear solution and time to fix the defect in time.
- There are always solutions available to prevent defects at any time.

2.2.2 Reviewing:

- They often say 'honesty is the best policy' and quality reviews rely on this. Therefore, the person in charge needs to be honest, unbiased.
- Based on the defects, it should be reviewed and prioritised for repair. And the details are recorded by tracking software.
- When a defect is discovered, it is necessary to notify the person responsible for the defect and find a solution to correct the defect as quickly as possible.

- After assessing and correcting the defect, the person in charge should give good advice to the members.

2.2.3 Unit Testing:

- Test cases are carefully prepared to suit each function of the system. When testing, the person in charge needs to prepare for bad cases to happen. The accuracy of the test is high, no cases are missed.
- Defects are assessed by severity and defects should be recorded using bug tracking software.
- When a defect is found, it is necessary to immediately notify the responsible person and propose a plan to repair the defect as soon as possible.

2.2.4 Integration Testing:

- Test cases need to be carefully prepared to fit the functionality of the system. Ready to repair and replace when bad things happen. High accuracy, no missed test cases.
- Defects should be assessed in terms of importance. Defects detected should be recorded with bug tracking and remediation software.
- When testing, if a defect is detected, it must be reported immediately to the responsible person and propose a repair plan as soon as possible.
- Internal modules within the system work smoothly.

2.2.5 System Testing:

- Test cases need to be carefully and accurately prepared to suit the system. Test cases must match the project's system and architecture. Do not miss any test cases.
- Depending on the severity of the defect and priority. A reasonable solution should be provided to correct the defect. Defect tracking and management with bug tracking software.
- When detecting a defect, it is necessary to notify the person responsible for the defect. It is necessary to have solutions to repair defects as soon as possible.

2.3 Training Plan

Training Area	Participants	Duration	Waiver Criteria
Process healthcare	All team members	10/01/2022 - 15/01/2022	Mandatory
Bootstrap, Html 5 (Front-end for web)	VietH	16/01/2022 - 13/02/2002	Mandatory

SpringBoot, Rest API (Back end for web)	QuanVV, LinhNH	16/01/2022 - 13/02/2022	Mandatory
Coding Convention and Bug Logging Convention	All team members	16/01/2022 - 13/02/2022	Mandatory
Github	All team members	16/01/2022 - 27/01/2022	Mandatory

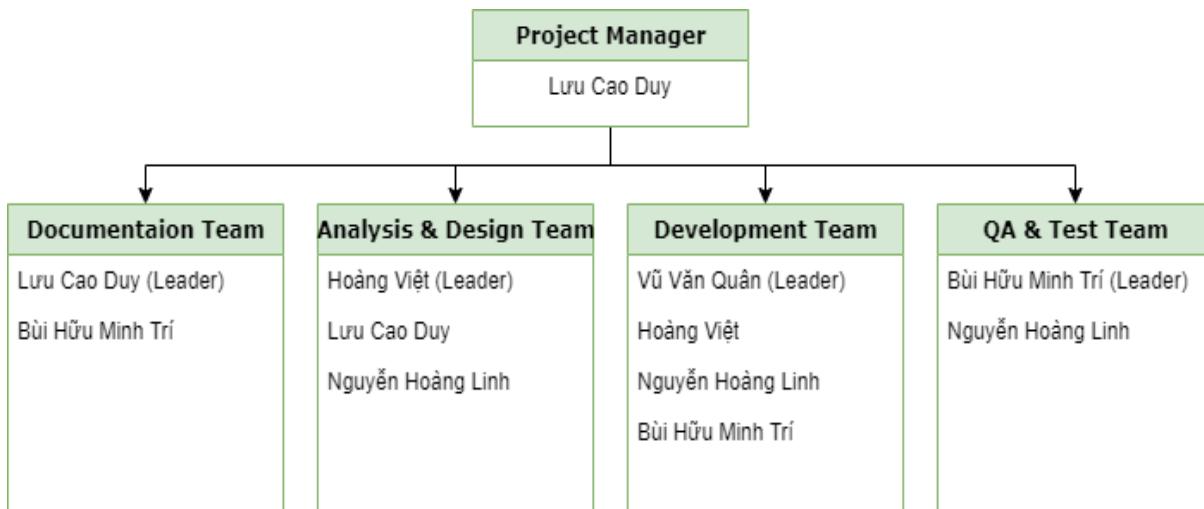
3. Project Deliverables

#	Deliverable	Due Date	Deliverable Scope
0	Project introduction	10/1/2022	Overview Project, Policy, Project Charter
2	Project Plan	18/1/2022	Time Management, Risk Management, Resource Management
3	SRS for Integration 1	25/1/2022	User Requirement, User Case, Use Case Specification, ERD
4	SSD for Integration 1	13/2/2022	Architecture Design, Detailed Design, Database
5	SSD for Integration 2	3/3/2022	Architecture Design, Detailed Design, Database
6	SRS for Integration 2	18/3/2022	Architecture Design, Detailed Design, Database
7	SSD for Integration 3	25/2/2022	User Requirement, User Case, Use Case Specification, ERD

8	SRS for Integration 3	15/3/2022	User Requirement, User Case, Use Case Specification, ERD
10	Code Package 1	14/2/2022	Code & Unit test, System test cases
12	Code Package 2	9/3/2022	Code & Unit test, System test cases
14	Code Package 3	25/3/2022	Code & Unit test, System test cases
15	Code Package 4	5/4/2022	Code & Unit test, System test cases
16	Testing document	15/4/2022	System and acceptance test reports
17	User Guides Document	18/4/2022	User manuals
18	Project Final Report	22/4/2022	All of information in reports
19	Final Package	23/4/2022	Final Codes & documents, User manual

4. Responsibility Assignments

4.1 Team & Structures:



4.2 Roles & Responsibilities:

Role	Responsibility
Project Manager	<ul style="list-style-type: none">- Planning and defining scope- Developing schedule- Coordinating communication- Reviewing and tracking document- Assign individual responsibility- Assign task to team members
Documentation Leader	<ul style="list-style-type: none">- Reviewing and tracking requirement- Tracking and completing report documents
Documentation Member	<ul style="list-style-type: none">- Create report documents

	<ul style="list-style-type: none"> - Creating SRS, SDS, Final Document
Analysis & Design Leader	<ul style="list-style-type: none"> - Analysis user requirement - Reviewing and tracking design - Identify main UI and design screen - Draw prototype, architecture design - Design database
Analysis & Design Member	<ul style="list-style-type: none"> - Analysis user requirement - Design database - Draw architecture design - Do details design
Development Leader	<ul style="list-style-type: none"> - Developing project - Solve issue of project
Development Member	<ul style="list-style-type: none"> - Develop Front-end - Develop Back-end - Develop ETL core
QA & Test Leader	<ul style="list-style-type: none"> - Execute unit testing - Create quality management plan - Create testing plan - Responsible for test execution - Create quality and test report

	<ul style="list-style-type: none"> - Review testcase - Assess quality of project
QA & Test Member	<ul style="list-style-type: none"> - Create test case - Execute test

5. Project Communications

5.1 Communication Plan

Communication Item	Who/ Target	Purpose	When, Frequency	Type, Tool, Method(s)
Team weekly meeting	All members and lecturer	Review and determine next week's job	20:30 every Wednesday, Sunday	Online through Facebook, Zalo, Google Meet
Team Daily Meeting	All members	Check and report the progress of the members every day	9:00 pm to 9:30 pm everyday	Facebook, Google Meet
Team daily working	All members	Work together	9:00 am to 9:00 pm everyday	Google Meet
Unscheduled meeting	All members	During the project process, someone who has an important problem needs to be solved or needs help, discussing then solve that problem.	There are some important things when found by members.	Facebook, Zalo, Google Meet

5.2 External Interface

5.2.1 FU Contacts

Function	Contact Person (Name, Position)	Contact Address (Email, Phone Number)	Responsibility
Supervisor	Tran Thu Thuy	Thuytt108@fe.edu.vn 090 4380878	<ul style="list-style-type: none"> - Provide document template - Instruct the project team - Review deliverables - Supervise project status - Receive report project

5.2.2 Customer Contacts

No customer yet.

6. Configuration Management

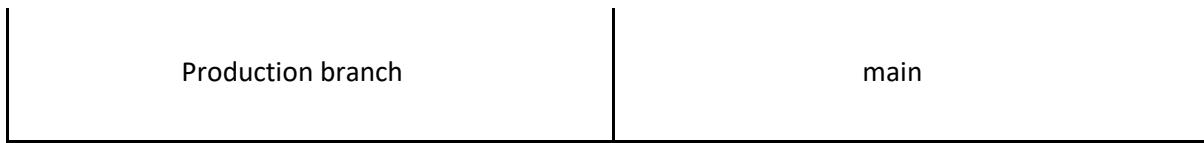
6.1 Document Management

- Document tools: Microsoft Word 2016, Microsoft Excel 2016.
- File management: Google Drive.

6.2 Source Code Management

- Source code is managed by Git on github.com
- Branching model:

Category	Branch Name/ Prefixes
Development branch	dev
New feature branch	feat/
Fixing bug branch	bugfix/



6.3 Tools & Infrastructures

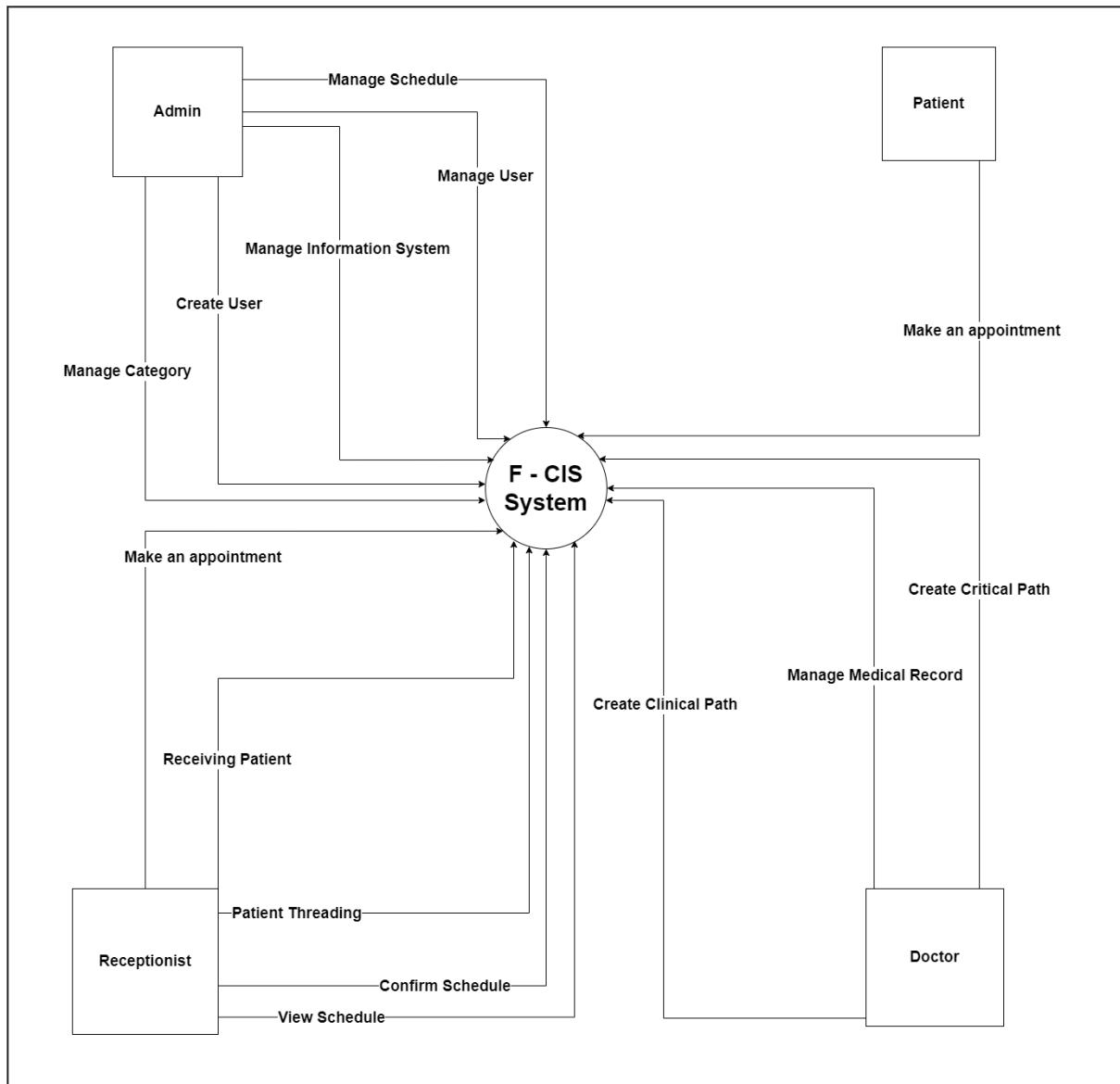
Category	Tools & Techniques
Programming languages	Javascript, Java
Framework	Bootstrap, Spring boot
API	REST API
API management tool	Postman
DBMS	MySQL
IDEs/Editors	Visual Studio Code, IntelliJ
UML tools	Draw.io
Version Control	GitHub

III. Software Requirement Specification

1. Product Overview

This chapter outlines the functional and non-functional requirements of our system. It also provides some format constraints in common requirements and project success criteria. The content of this chapter is used as the basis for the work in the subsequent chapters.

In recent years, the healthcare industry is receiving more attention and investment development, our CIS system will contribute to helping newly established medical facilities strengthen their management and Treat patients faster and easier.



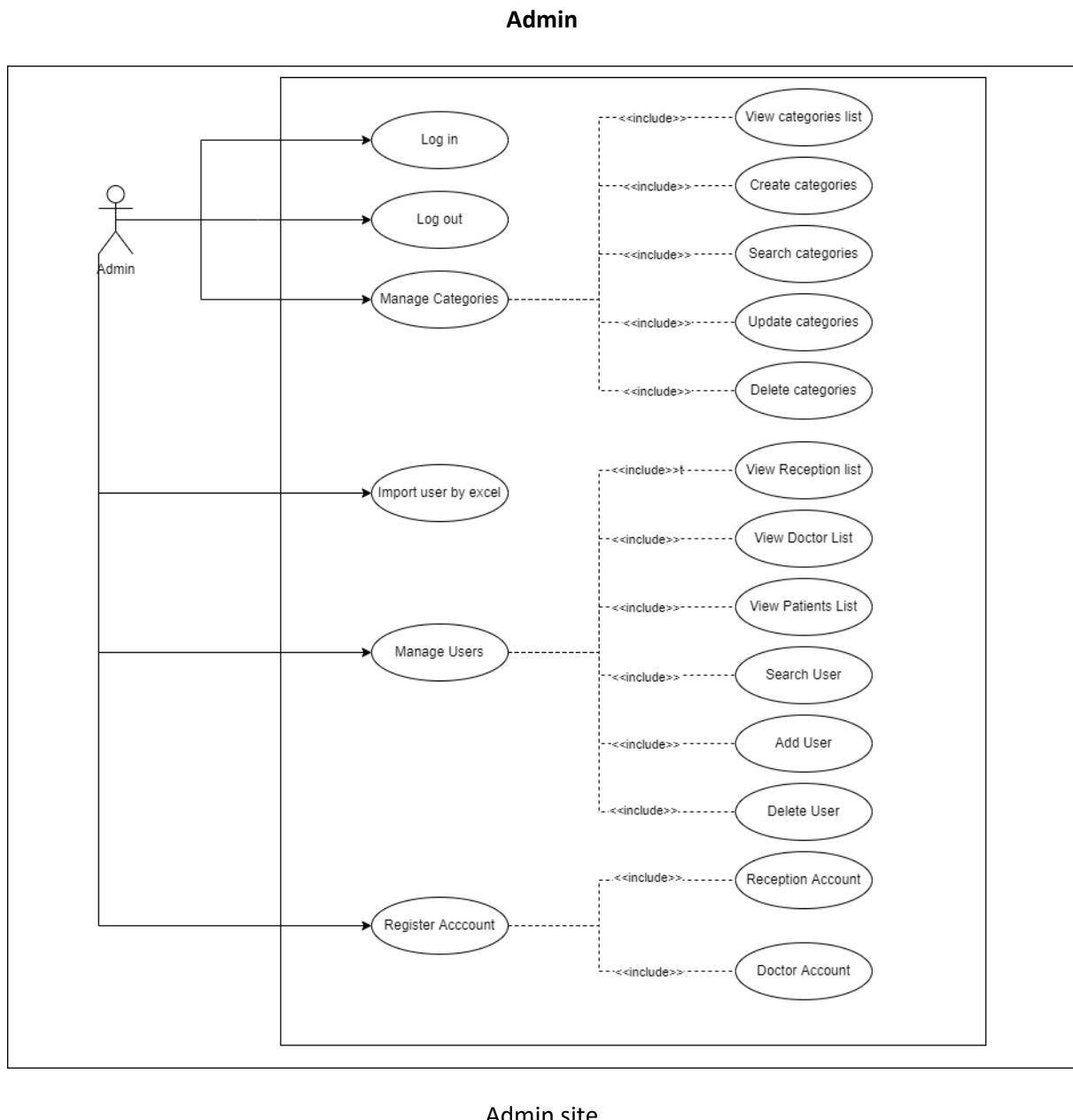
2. User Requirements

2.1 Actors

#	Actor	Description
1	Admin	Manager of software who create, update, archive account, set role for doctor and receptionist
2	Receptionist	Who receiver patient's request, threading patients
3	Doctor	Who evaluation and diagnosis of disease, make clinical path, give medical instruction
4	Patient	Who use the system to make an appointment

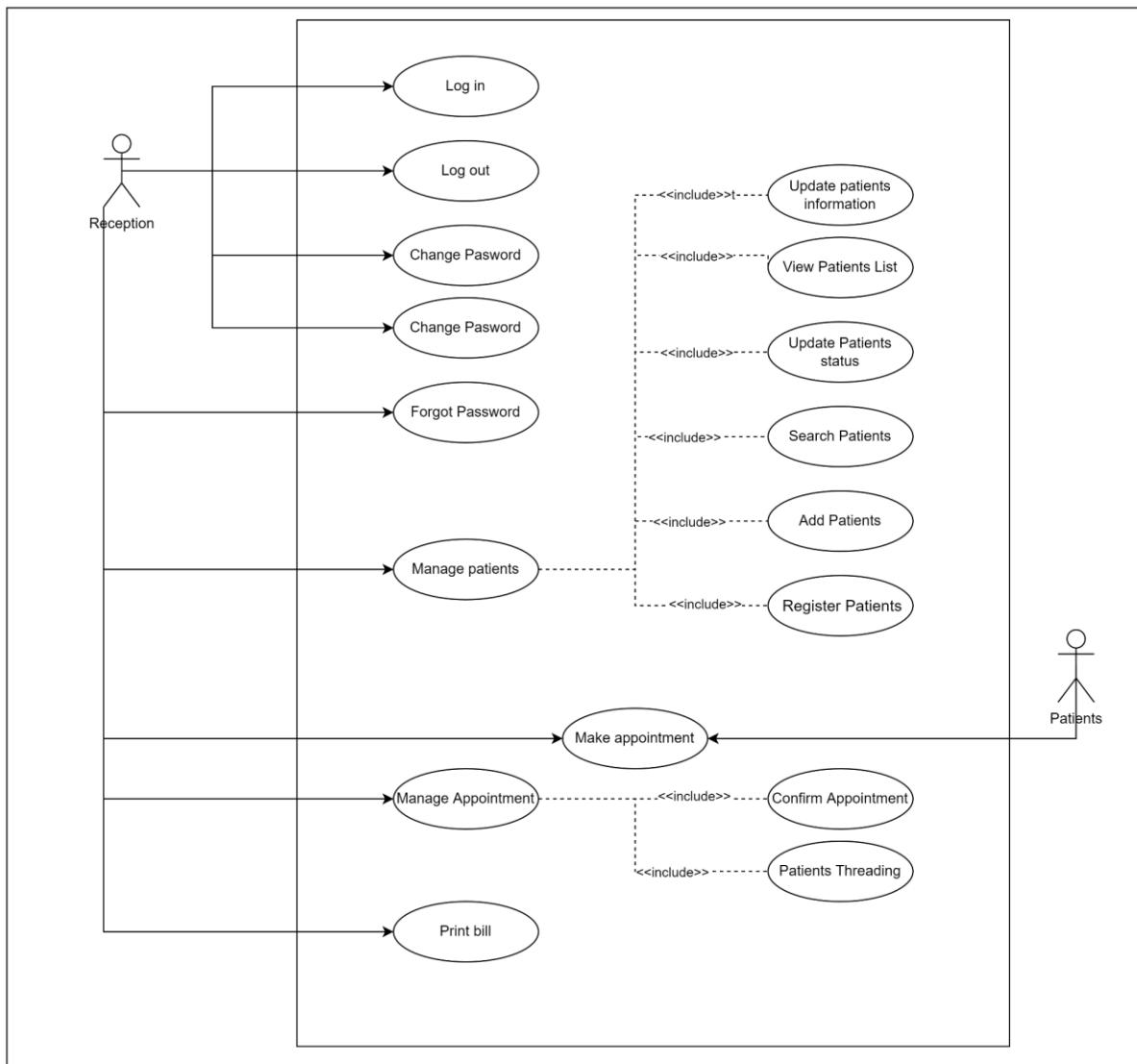
2.2 Use Cases

2.2.1 Diagrams



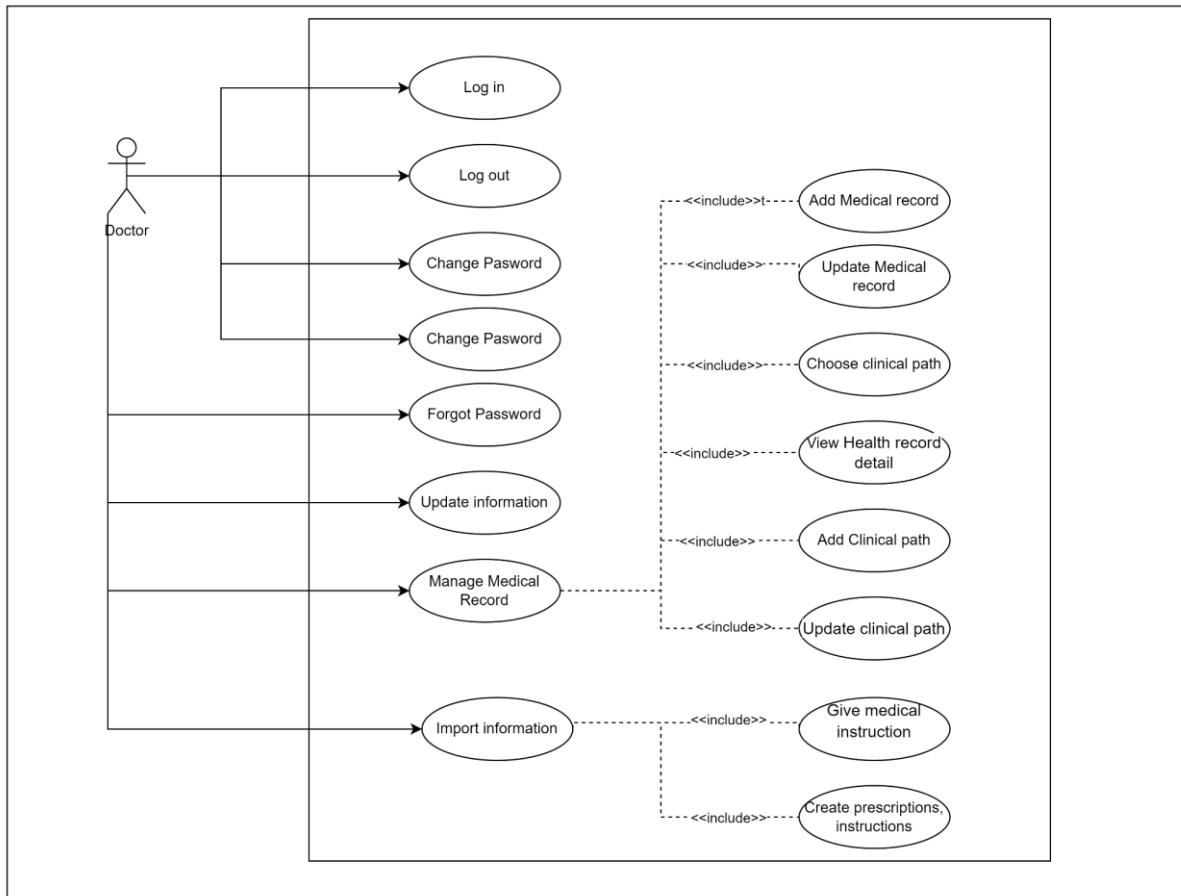
Admin site

Reception and patients



Reception and patient site

Doctor



Doctor site

2.2.2 Use Case List

ID	Use Case	Primary Actors	Secondary Actor
UC-1	View Categories	Admin	N/A
UC-2	Search Categories	Admin	N/A
UC-3	Create Categories	Admin	N/A
UC-4	Delete Categories	Admin	N/A

UC-5	Edit Categories	Admin	N/A
UC-6	Login	Admin	N/A
UC-7	Logout	Admin	N/A
UC-8	View Doctors	Admin	N/A
UC-9	Search Doctors	Admin	N/A
UC-10	Create Doctors	Admin	N/A
UC-11	Delete Doctors	Admin	N/A
UC-12	Edit Doctors	Admin	N/A
UC-13	View Receptionist	Admin	N/A
UC-14	Search Receptionist	Admin	N/A
UC-15	Create Receptionist	Admin	N/A
UC-16	Delete Receptionist	Admin	N/A
UC-17	Edit Receptionist	Admin	N/A
UC-18	View Patients	Admin	N/A

UC-19	Create Accounts	Admin	N/A
UC-20	Delete Accounts	Admin	N/A
UC-21	View Schedule	Admin	N/A
UC-22	Create Schedule	Admin	N/A
UC-23	Login	Doctor	N/A
UC-24	Log out	Doctor	N/A
UC-25	Change Password	Doctor	N/A
UC-26	View Medical Record	Doctor	N/A
UC-27	Create Medical Record	Doctor	N/A
UC-28	Delete Medical Record	Doctor	N/A
UC-29	Edit Medical Record	Doctor	N/A
UC-30	View Patient Health Record History List	Doctor	N/A
UC-31	View Patient Health Detail Record	Doctor	N/A

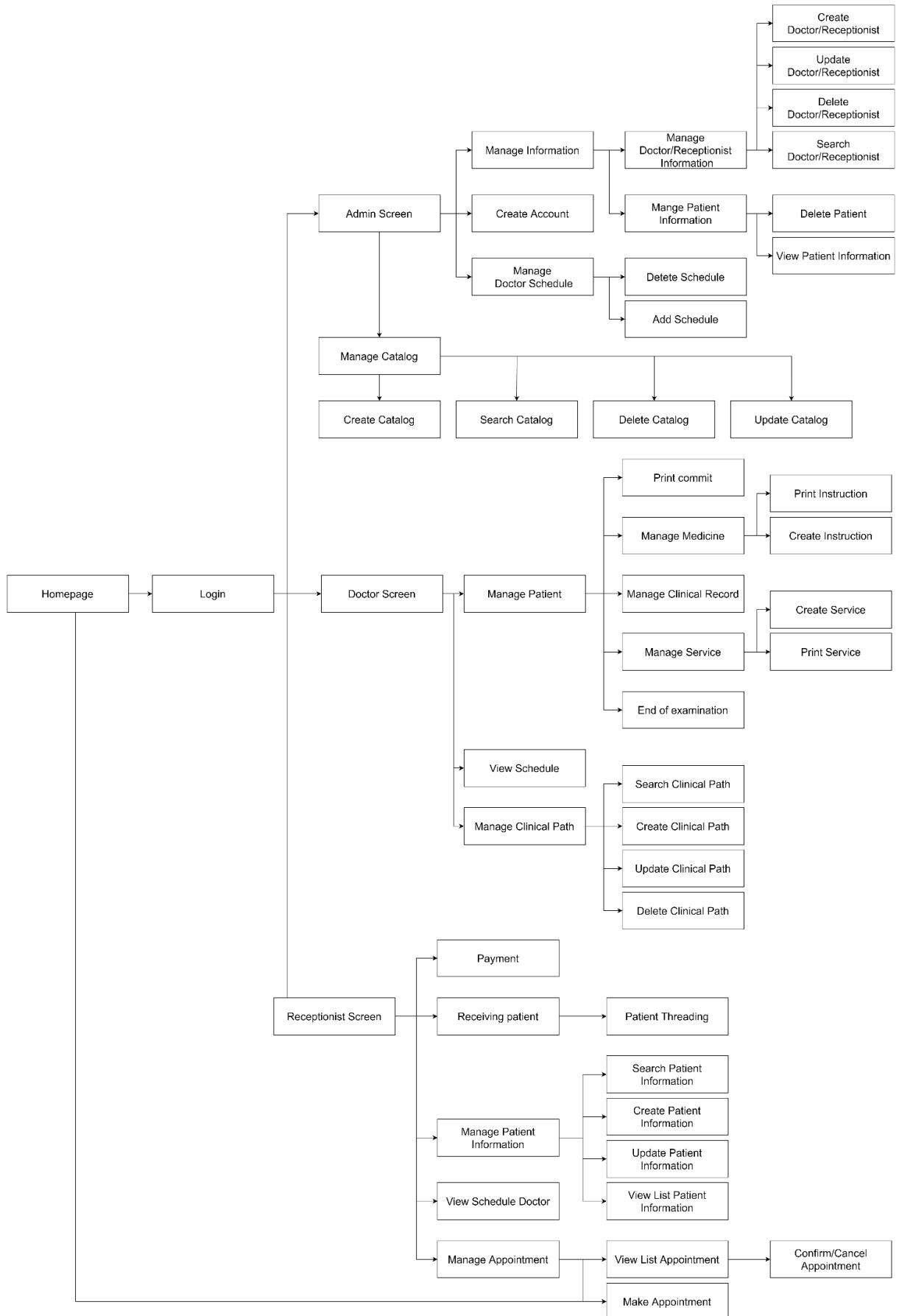
UC-32	Enter Examination Information	Doctor	N/A
UC-33	Create Medical Issue	Doctor	N/A
UC-34	Create Prescriptions	Doctor	N/A
UC-35	View List Clinical Path	Doctor	N/A
UC-36	Add clinical Path	Doctor	N/A
UC-37	Update Clinical Path	Doctor	N/A
UC-38	Login	Receptionist	N/A
UC-39	Logout	Receptionist	N/A
UC-40	Change Password	Receptionist	N/A
UC-41	View Doctors	Receptionist	N/A
UC-42	Search Doctors	Receptionist	N/A
UC-43	Make Appointment	Receptionist	N/A
UC-44	Check Appointment	Receptionist	N/A
UC-45	Cancel Appointment	Receptionist	N/A

UC-46	View Patients	Receptionist	N/A
UC-47	Search Patients	Receptionist	N/A
UC-48	Print Invoices And Prescriptions	Receptionist	N/A
UC-49	Threading Patient	Receptionist	N/A
UC-50	Make Appointment	Patient	N/A

3. Functional Requirements

3.1 System Functional Overview

3.1.1 Screens Flow



3.1.2 Screen Descriptions

	Feature	Screen	Description
1	Authentication	Login	Screen for admin, receptionist and doctor to sign in into the system.
2	Authentication	Forget	Screen for admin, receptionist and doctor to change the new password when them forget password.
3	Authentication	Change password	Screen for admin, receptionist and doctor to change the password.
4	Make Appointment	Appointment	Screen for Receptionist, patient make an appointment.
5	Appointment list	Appointment list	Manage appointment list, can search by appointment date, appointment status, name, phone number.
6	Appointment confirmation	Appointment Detail	<p>Select to view appointment details from the appointment list screen.</p> <p>The Receptionist will call the patient to confirm the appointment, in case the patient does not exist in the system, generate a patient code and confirm the appointment.</p> <p>Can do Cancel Appointment.</p>
7	Receiving patients	Reception	When the patient directly comes to book an appointment at the counter, if it is a new person, the receptionist will create new information for the patient, if it is an old patient, the receptionist will check the available information from the database.
8	New patient registration	Patient Register	Create a new medical record for new patient.
9	Make an appointment for the patient	Patient Appointment	<p>Patients cannot make an appointment on the internet, so they call the clinic directly to make an appointment.</p> <p>Receptionist relies on patient information to create an appointment.</p>

10	View Doctor Information And Schedule	Doctor Schedule	Based on doctors' status and appointment schedule, create weekly work schedule for doctors.
11	Manage Patient Information	Patient information	View, add, edit, delete and search patient information.
12	Manage Doctor Information	Doctor information	View, add, edit, delete and search doctor information.
13	Manage Account	List Account	Create account access for doctors and receptionists.
14	Manage Medical Record (Add, update, choose clinical path)	Add and update medical record	View and search the patient's medical record, edit examination and treatment information in the patient's medical record.
15	Manage Medical Record	Patient Health	Detail medical record and medical history.
16	Manage the list of patients in the clinic	Clinical Station	List of patients sorted.
17	Create Clinical Path, treatment regimen	Add and update clinical path	Create Clinical Path, treatment regimen for patient, enter that to Medical record.
18	Patient threading	Patient threading	The receptionist sends the patient's records to the waiting list of each clinic in the order of priority, pre-booking and direct registration.
19	Manage categories	Manage categories	Manage categories that the clinic has, including technical services, test codes, disease codes, drug codes.

3.1.3 Screen Authorization

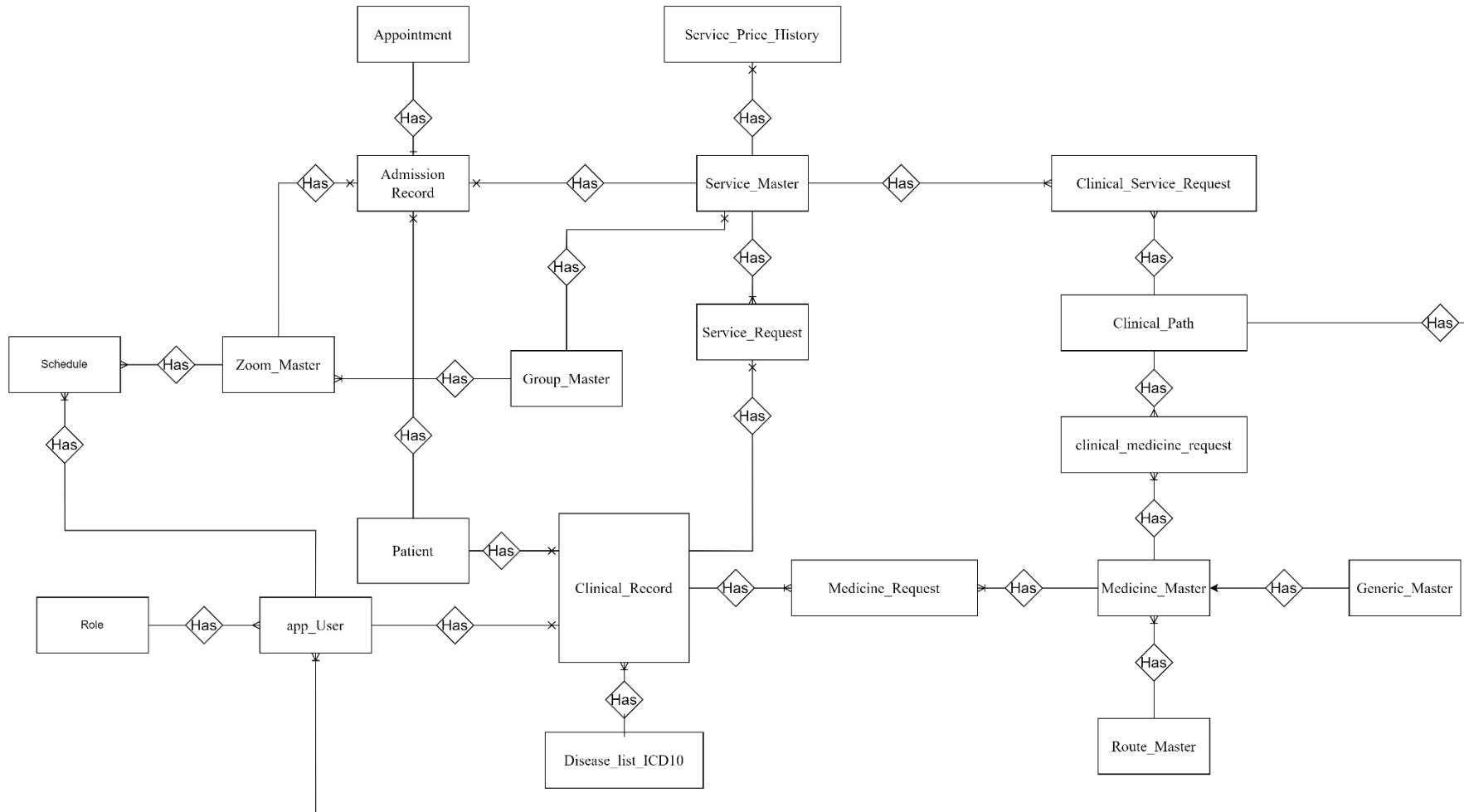
Screen	Admin	Receptionist	Doctor	Patient
Home		X	X	X
Log in		X	X	
Forgot Password	X	X	X	
Appointment				X
Appointment List		X		
Appointment Detail		X		
Reception		X		
Patient Register		X		
Patients Appointment		X		
Manage Patients	X	X		
Manage Doctor	X			
Manage Receptionist	X			
Manage Schedule	X			
Manage medical record			X	
Patient Health record history list			X	
Patient Health detail record			X	

Clinical Station			X	
Examination			X	
Medical instruction			X	
Clinical path			X	
Manage categories	X			

3.1.4 Non-Screen Functions

#	Feature	System Function	Description
1	Home	Sign out	Users sign out of the system.
2	Print invoices and prescriptions	Print	Reception prints the bill and prescription

3.1.5 Entity Relationship Diagram

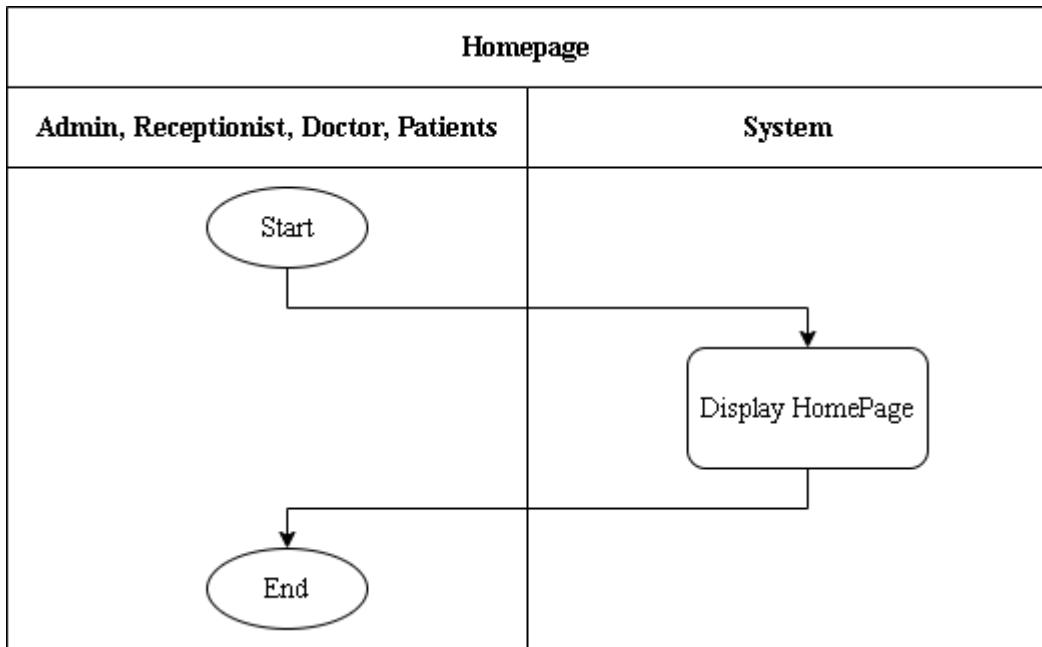


Entities List

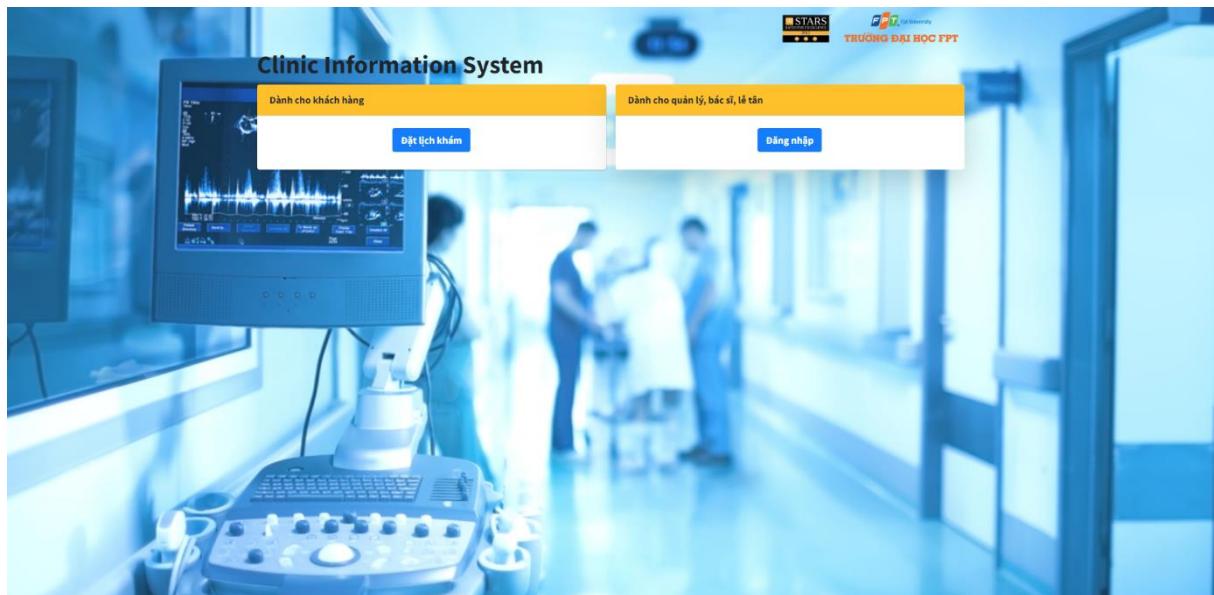
	Entity	Description
1	Admin	The admin of F-CIS system.
2	Doctor	The account of the F-CIS system.
3	Receptionist	The account of the F-CIS system.
4	Patient	The customer who uses the F-CIS system to make appointment.
5	Manage account	Describe display all account.
6	Create patient medical record	Describe the receptionist creating a medical record for a patient.
7	Create receptionist account	Describe the admin creating an account for the receptionist.
8	Create doctor account	Describe the admin creating an account for the doctor.
9	Update medical record	Describe doctor update medical record of patient.
10	Manage medical record	Describe display all medical record of patient by admin.
11	Clinical Path, treatment regimen	Clinical Path, treatment regimen which create by doctor for each patient.

3.2 Homepage

The following figure show how this function is triggered:



Screen layout:



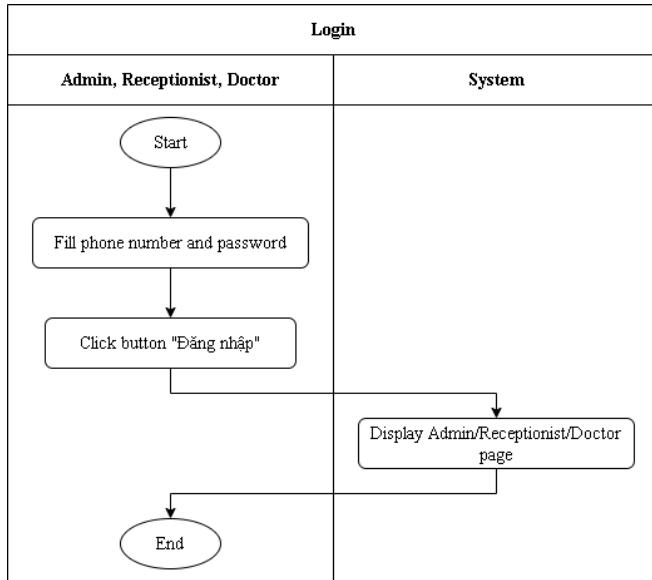
Functional Requirement:

- Everyone can view the Homepage.

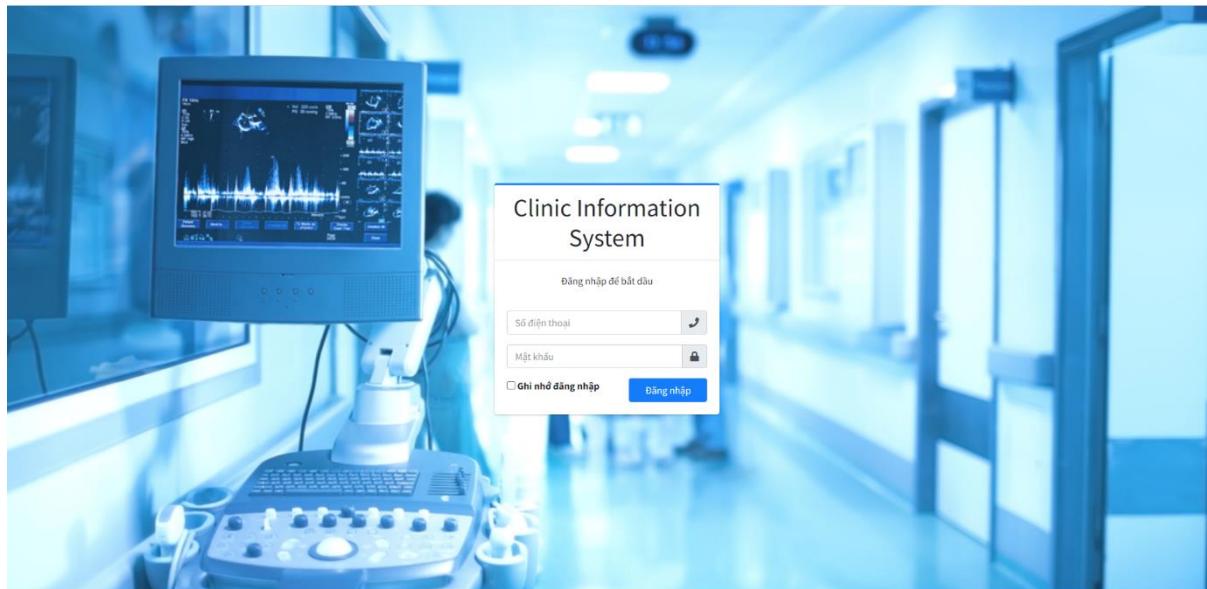
3.3 Authentication

3.3.1 Login

The following figure show how this function is triggered:



Screen layout:



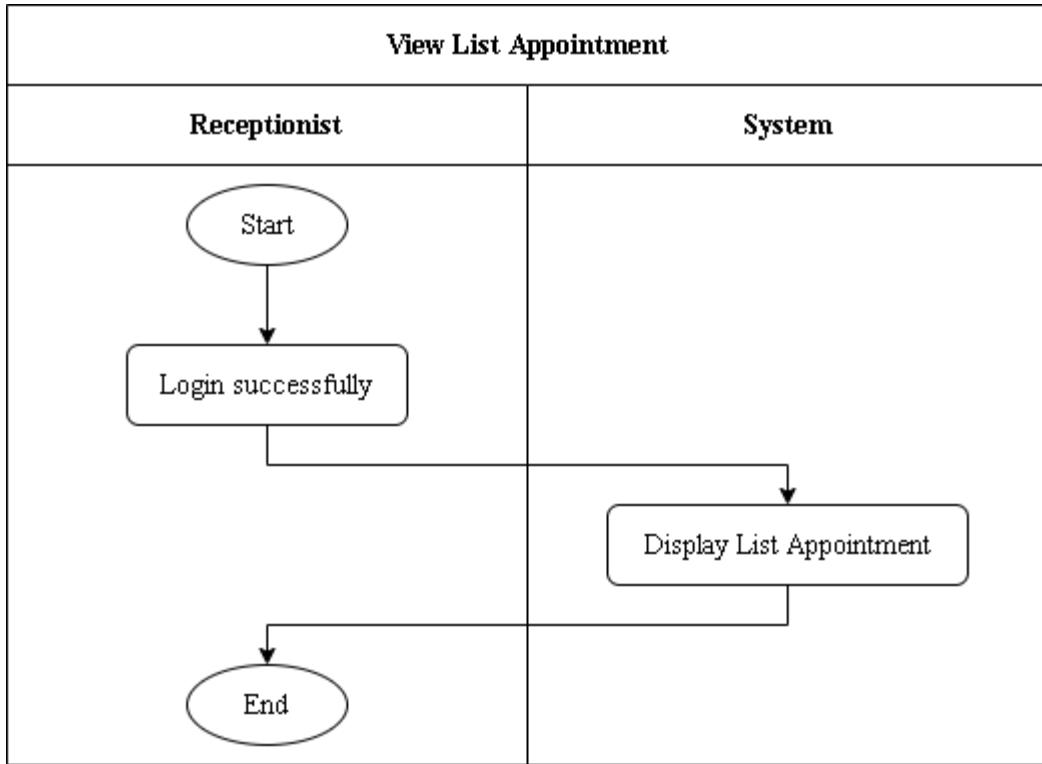
Functional Requirement:

- Validate phone number and password of account in software.
- Account is created by admin.

3.4 Manage Appointment

3.4.1. View List Appointment

The following figure show how this function is triggered:



Screen layout:

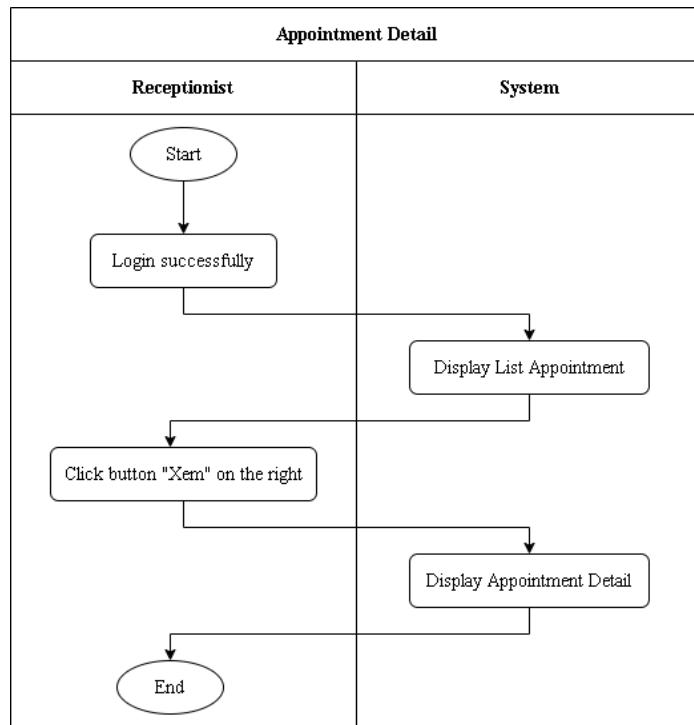
#	Mã bệnh nhân	Tên	Email	Điện thoại	Ngày sinh	Giới tính	Actions
1	75	Bùi Thị	tbtm14@gmail.com	0389617414	02/03/1999	Nam	
2	72	Võ Hường	viettt12@gmail.com	0389617412	09/03/2001	Nam	
3	76	Cao Duy	duycau12@gmail.com	0389617413	11/12/1999	Nam	
4	77	Tô Văn Thúy	thuytovan12@gmail.com	0389617415	04/08/1999	Nữ	
5	93	Lê Tuấn Anh		0355556674	10/01/1997	Nam	
6	92	Hoàng Linh	hoanglinh12@gmail.com	0309782000	01/01/1999	Nam	

Functional Requirement:

- Receptionist can view the list appointment.

3.4.2 Appointment Detail

The following figure show how this function is triggered:



Screen layout:

The screenshot shows a web-based form titled 'ĐẶT LỊCH KHÁM'. The form fields include:

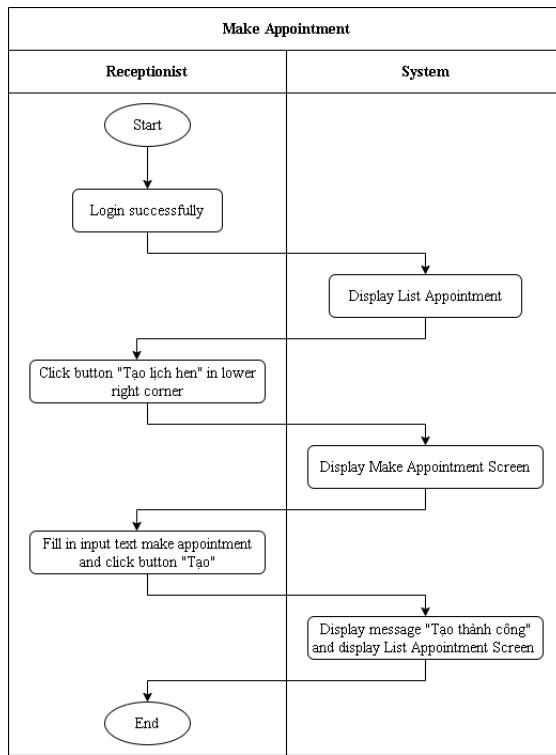
- Số điện thoại (*) - Input: 0904380878
- Họ và tên (*) - Input: Trần Thu Thủy
- Ngày sinh (*) - Input: 02/05/1980
- Địa chỉ - Input: 136 Hồ Tùng Mậu
- Chọn ngày khám (*) - Input: 25/04/2022
- Chọn giờ khám (*) - Input: (dropdown menu)
- Vấn đề sức khỏe cần khám - Textarea
- Mã bệnh nhân - Input: 110
- Giới tính (*) - Input: Nam
- Email - Input: (empty)
- Tên người Giám Hộ(nếu có) - Input: (empty)
- Số điện thoại Giám Hộ(nếu có) - Input: (empty)
- Mối quan hệ với người bệnh(nếu có) - Input: (empty)
- Chọn phòng - Input: Phụ Sản
- Trạng thái: Đang chờ
- Action buttons at the bottom: Xác nhận lịch hẹn (green), Tiếp nhận (blue), Hủy Lịch hẹn (yellow), Quay lại (orange).

Functional Requirement:

- Receptionist can view the appointment details.

3.4.3 Create New Appointment

The following figure show how this function is triggered:



Screen layout:

The screenshot shows a web-based form titled 'ĐẶT LỊCH KHÁM'. The form fields include:

- Số điện thoại (*)
- Họ và tên (*)
- Giới tính (*)
- Ngày sinh (*)
- Email
- Tên người Giám Hội (nếu có)
- Địa chỉ
- Số điện thoại Giám Hội (nếu có)
- Chọn ngày khám (*)
- Mỗi quan hệ với người bệnh (nếu có)
- Chọn giờ khám (*)
- Vấn đề sức khỏe cần khám
- Chọn phòng
- Trạng thái:

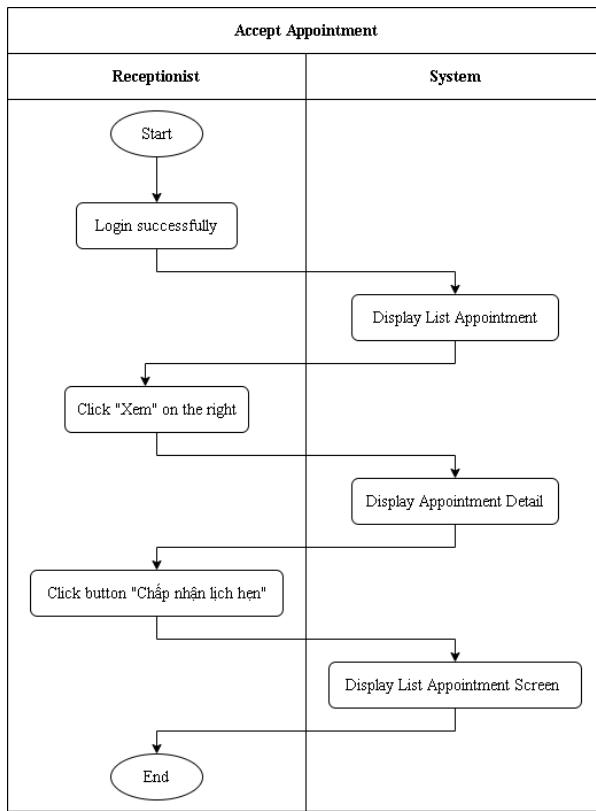
At the bottom of the form are three buttons: 'Xác nhận lịch hẹn' (Confirm appointment), 'Tiếp nhận' (Accept), and 'Quay lại' (Back).

Functional Requirement:

- Receptionists and patients can make appointments.

3.4.4 Accept Appointment

The following figure show how this function is triggered:



Screen layout:

ĐẶT LỊCH KHÁM

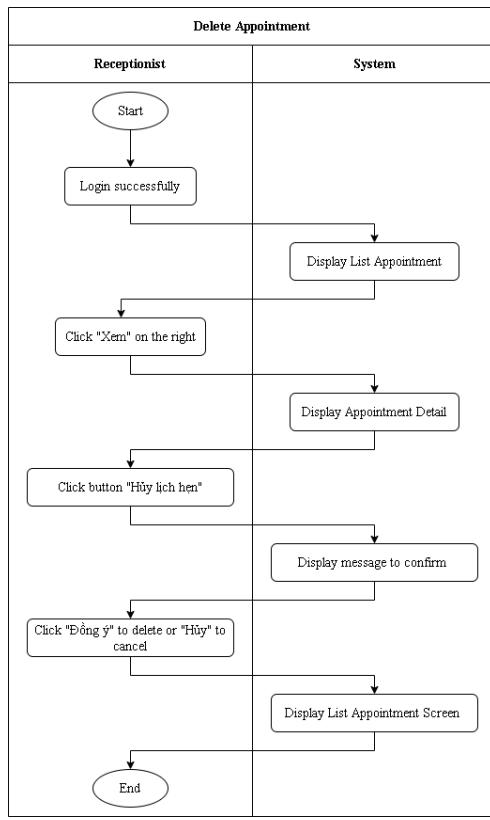
Số điện thoại (*) <input type="text" value="0904380878"/>	Mã bệnh nhân <input type="text" value="110"/>
Họ và tên (*) <input type="text" value="Trần Thu Thủy"/>	Giới tính (*) <input type="text" value="Nam"/>
Ngày sinh (*) <input type="text" value="02/05/1980"/>	Email <input type="text"/>
Địa chỉ <input type="text" value="136 Hồ Tùng Mậu"/>	Tên người Giám Hộ(nếu có) <input type="text"/>
Chọn ngày khám (*) <input type="text" value="25/04/2022"/>	Số điện thoại Giám Hộ(nếu có) <input type="text"/>
Chọn giờ khám (*) <input type="text"/>	Mối quan hệ với người bệnh(nếu có) <input type="text"/>
Vấn đề sức khỏe cần khám <input type="text"/>	
Chọn phòng <input type="text" value="Phụ Sản"/>	
Trạng thái: <input type="text" value="Đã xác nhận"/>	
<input type="button" value="Xác nhận lịch hẹn"/> <input type="button" value="Tiếp nhận"/> <input type="button" value="Hủy Lịch hẹn"/> <input type="button" value="Quay lại"/>	

Functional Requirement:

- Receptionists can accept appointments.

3.4.5 Delete Appointment

The following figure show how this function is triggered:



Screen layout:

The screenshot shows the 'ĐẶT LỊCH KHÁM' (Appointment Booking) form. It includes fields for personal information (Số điện thoại, Họ và tên, Ngày sinh, Địa chỉ), medical details (Chọn ngày khám, Chọn giờ khám, Vấn đề sức khỏe cần khám), and contact information for a guardian (Mã bệnh nhân, Giới tính, Email, Tên người Giám hộ, Số điện thoại Giám hộ). At the bottom, there are buttons for 'Chọn phòng' (Select room), 'Trạng thái' (Status - set to 'Đã hủy'), and two action buttons: 'Hủy Lịch hẹn' (Cancel Appointment) and 'Quay lại' (Back).

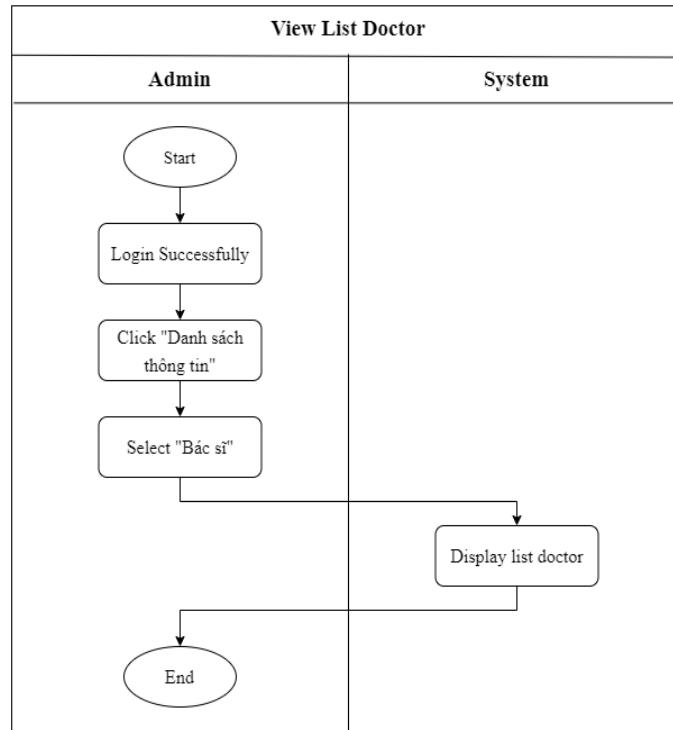
Functional Requirement:

- Receptionists can delete appointments.

3.5 Manage Doctor Information

3.5.1 View List Doctor

- The following figure show how this function is triggered:



Screen layout:

The screenshot shows a user management interface with the following details:

- Header:** CIS, Đăng xuất, System Admin, Quản lý người dùng, Type your keywords here, Trang chủ / Người dùng, Thêm người dùng.
- Left Sidebar:** Danh sách thông tin, Bác sĩ, Bệnh nhân, Lê tân, Quản trị viên, Lịch trực, Lịch tra/c, Danh mục, Danh mục dịch vụ kĩ thuật, Danh mục bệnh IDC10, Danh mục thuốc.
- Table:** A grid displaying user information with the following columns: #, Mã bệnh nhân, Tên, Email, Điện thoại, Ngày sinh, Giới tính. The data is as follows:

#	Mã bệnh nhân	Tên	Email	Điện thoại	Ngày sinh	Giới tính
1	48	Hoàng Việt	doctor@gmail.com	0323232322	10/10/2002	Nam
2	49	Nguyễn Tuấn Hoàng	hoangnt@gmail.com	0983500442	10/02/1974	Nam
3	50	Hoàng Quốc Bảo	baohq@gmail.com	0958456378	11/03/1988	Nam
4	51	Hoàng Đức Khải	khaihd@gmail.com	0356954781	12/03/1977	Nam
5	52	Đặng Hồng Giang	gianghd@gmail.com	0355847531	01/03/1985	Nữ
6	54	Đặng Bá Thành	thanhbd@gmail.com	0759635894	11/03/1956	Nam

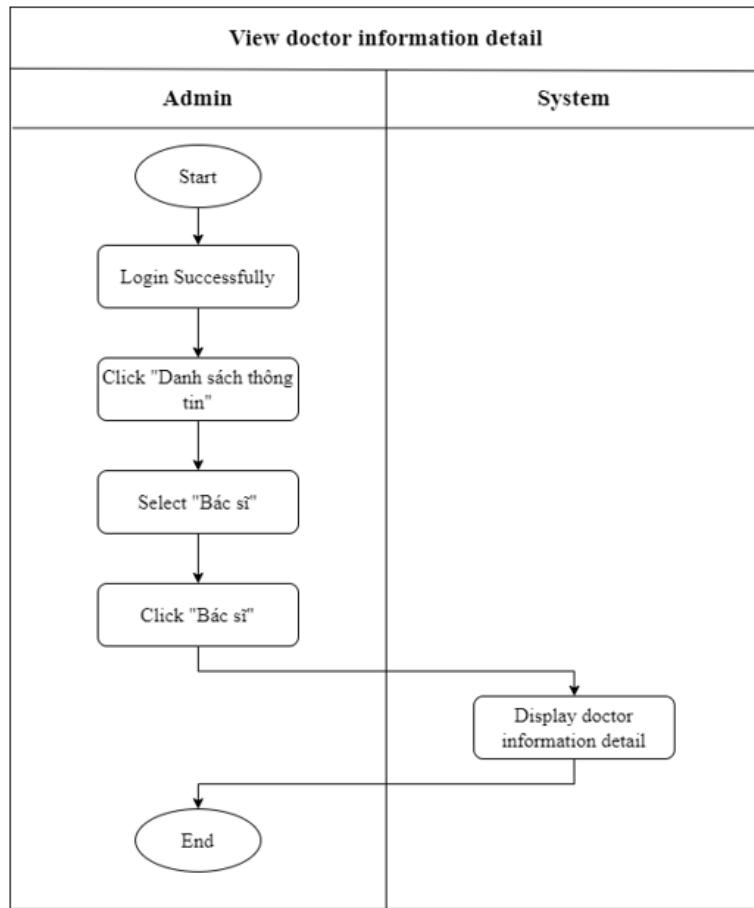
- Bottom:** Showing 1 to 6 of 20 entries, Go to page 1, Go to page 2.

Functional Requirement:

- Administrators can view the list of doctors.

3.5.2 Doctor Information Detail

The following figure show how this function is triggered:



Screen layout:

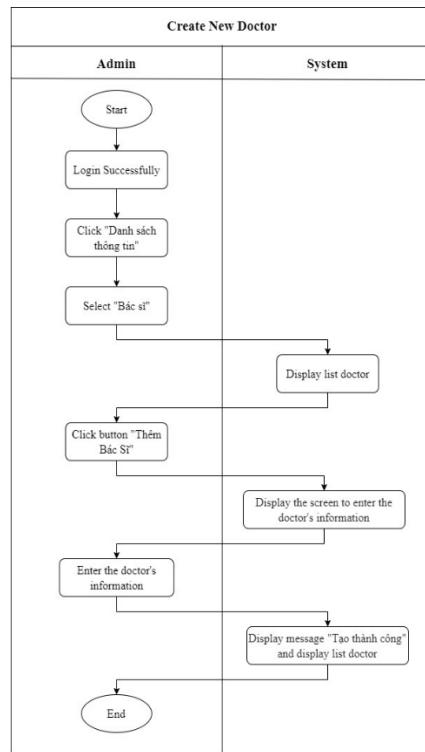
The screenshot shows a user management interface. On the left, there's a sidebar with navigation links like 'Danh sách thông tin', 'Bác sĩ', 'Điều nhân', 'Lê tân', 'Quản trị viên', 'Lịch trực', 'Lịch trực', 'Danh mục', 'Danh mục dịch vụ kĩ thuật', 'Danh mục bệnh IDC10', and 'Danh mục thuốc'. The main area is titled 'Quản lý người dùng' and shows a list of users with columns for 'Điện thoại', 'Ngày sinh', and 'Giới tính'. A modal dialog is open in the center, prompting for 'Điện thoại (*)' (Phone number), 'Mật khẩu (*)' (Password), 'Nhập lại mật khẩu (*)' (Re-enter password), 'Họ và tên (*)' (Name), 'Giới tính (*)' (Gender), 'Ngày sinh (*)' (Date of birth), 'Email address', and 'Address'. At the bottom of the modal are 'Hủy' (Cancel) and 'Lưu' (Save) buttons.

Function requirement:

- Admin view information of doctor.

3.5.3 Create New Doctor Information

The following figure show how this function is triggered:



Screen layout:

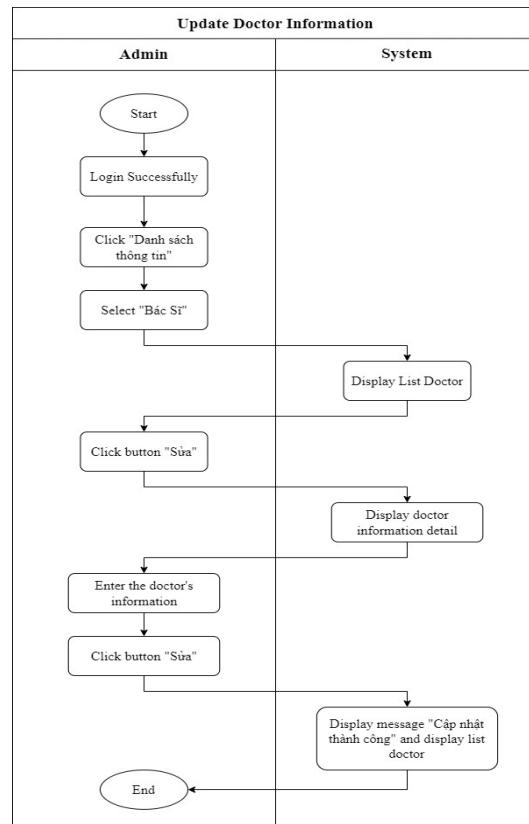
#	Mã bệnh nhân	Tên	Điện thoại	Ngày sinh	Giới tính
1	48	Hoàng Việt	0323232322	10/10/2002	Nam
2	49	Nguyễn Tuấn Hoài	0983500442	28/02/1974	Nam
3	50	Hoàng Quốc Bảo	0956456378	11/03/1988	Nam
4	51	Hoàng Đức Khải	0356954781	12/03/1977	Nam
5	52	Đặng Hồng Giang	0355847531	01/03/1985	Nữ
6	54	Đặng Bá Thành	0759635094	11/03/1956	Nam

Function requirement:

- Only admin can create account.
- Validate information when create account.

3.5.4 Update Doctor Information

The following figure show how this function is triggered:



Screen layout:

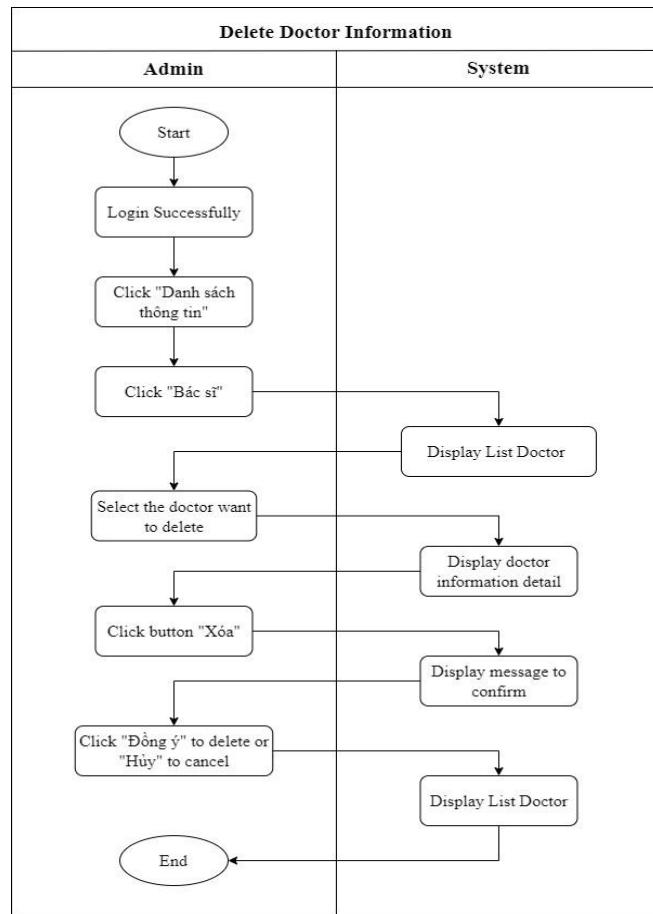
The screenshot shows the 'Quản lý người dùng' (User Management) screen. On the left, there is a sidebar with various menu items. The main area displays a table of users with columns: #, Mã bệnh nhân, Tên, Điện thoại (*), Ngày sinh, Giới tính, and actions (Edit, Delete). A modal window is open, allowing the update of a specific user's information. The modal fields include: Điện thoại (*), Mật khẩu (*), Nhập lại mật khẩu (*), Họ và tên (*), Giới tính (*), Ngày sinh (*), Email address, and Address. The user being updated has the following details: Điện thoại: 0323232322, Ngày sinh: 10/10/2002, Giới tính: Nam, Họ và tên: Nguyễn Tuấn Hải, and Email address: doctor@gmail.com.

Function requirement:

- Admin update information of doctor.
- Validate information when update information of doctor.

3.5.5 Delete Doctor Information

The following figure show how this function is triggered:



Screen layout:

The screenshot shows the 'Quản lý người dùng' (User Management) page. On the right, there is a modal dialog titled 'Delete User' with the question 'Bạn có chắc chắn xóa người dùng này không?'. Below the question are two buttons: 'Hủy' (Cancel) and 'Xóa' (Delete). The main table lists users with columns: #, Mã bệnh nhân, Tên, Email, Điện thoại, Ngày sinh, Giới tính, and actions (edit and delete icons). The table has 6 entries. At the bottom left, it says 'Showing 1 to 6 of 20 entries'. At the bottom right, there are navigation buttons for '1', 'Go', and arrows.

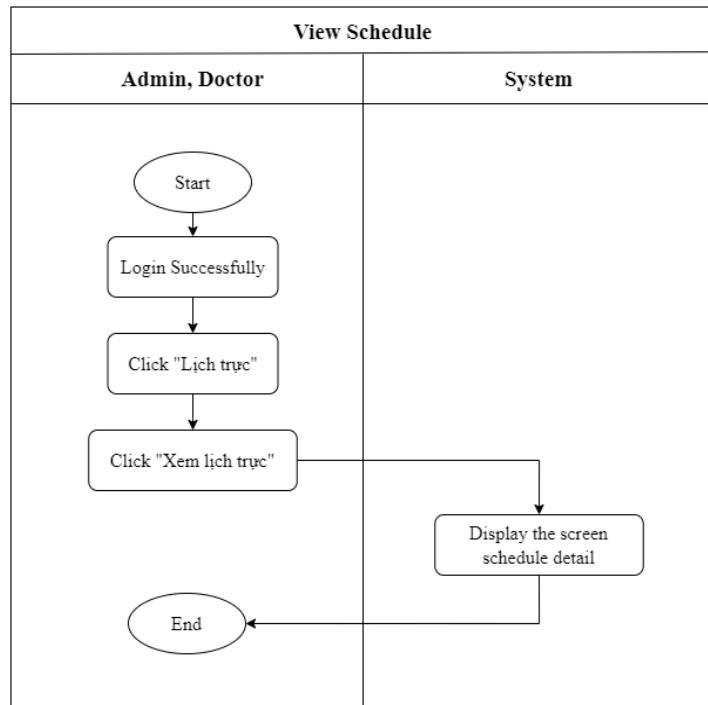
Function requirement:

- Only Admin can delete a doctor.

3.6 Manage Doctor Schedule

3.6.1 View Schedule

The following figure show how this function is triggered:



Screen layout:

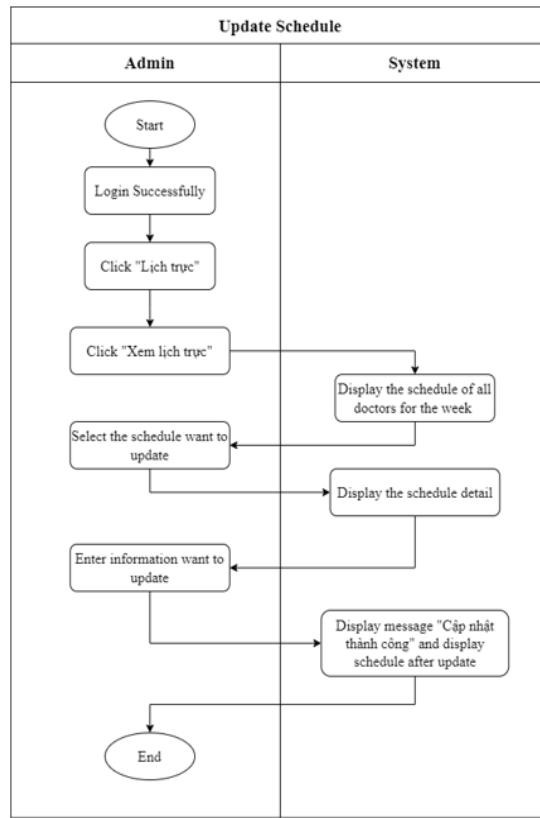
The screenshot shows the CIS application interface. On the left is a dark sidebar with navigation links: Danh sách thông tin, Bác sĩ, Bệnh nhân, Lễ tân, Quản trị viên, Lịch trực (highlighted), and Lịch trực. The main content area has a header 'Lịch trực'. It displays a table for the week starting from Monday (Thứ 2). The table rows are: Năm: 2022, Tuần: 3/1 to 7/1, Sáng 7h30 - 11h, and Chiều 14h - 17h30. A blue button labeled 'Tạo lịch trực' is visible at the top right of the table area.

Functional Requirement:

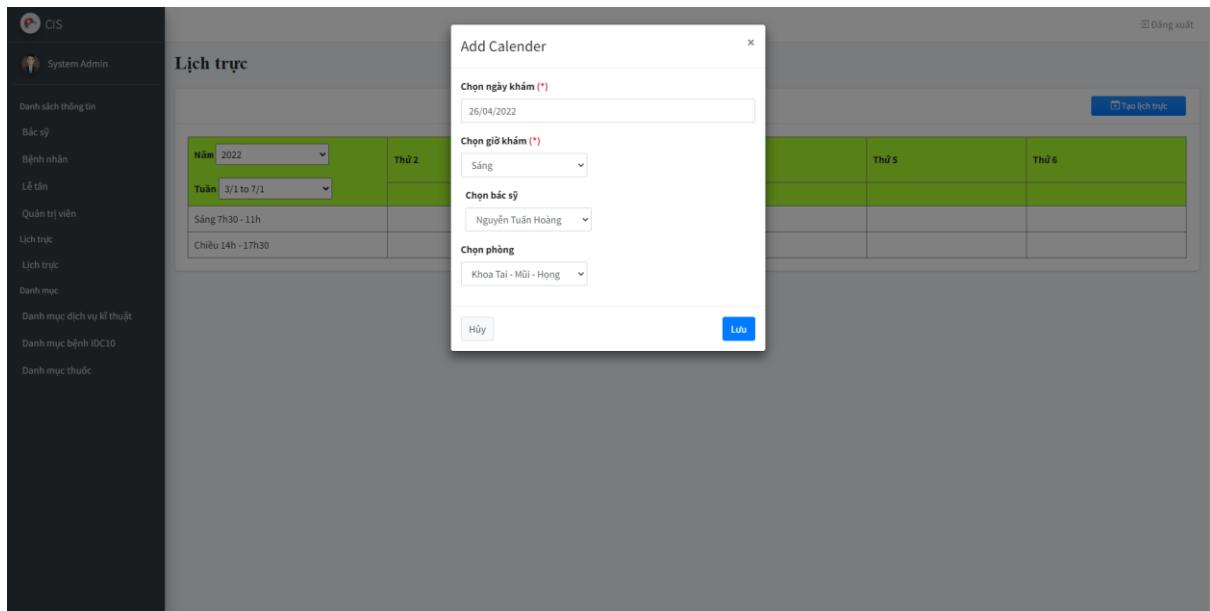
- Admin and doctor can view schedules.

3.6.2 Update Doctor Schedule

The following figure show how this function is triggered:



Screen layout:



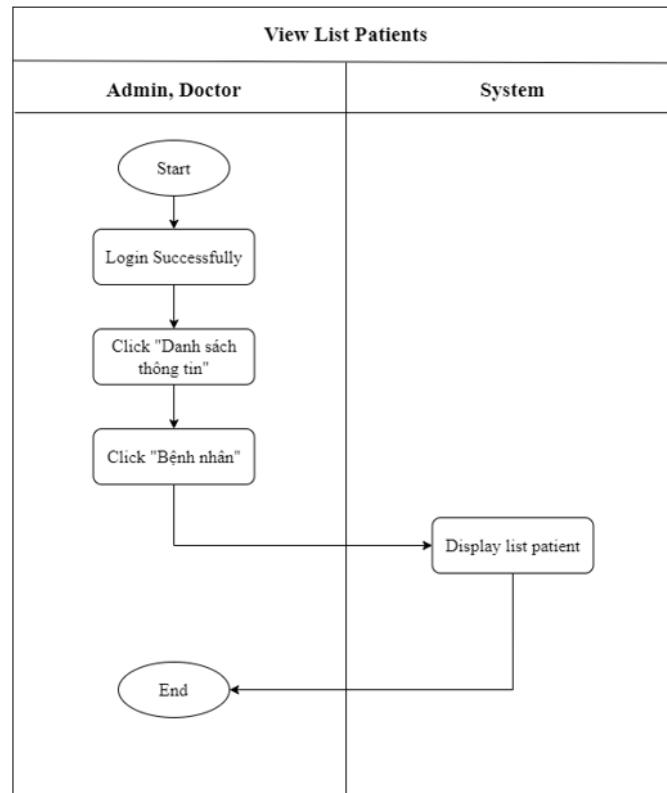
Functional Requirement:

- Admin can update doctor schedules.

3.7 Manage Patient Information

3.7.1 View List Patient

The following figure show how this function is triggered:



Screen layout:

The screenshot shows a user management interface. On the left is a sidebar with navigation links: "Danh sách thông tin", "Bác sĩ", "Bệnh nhân" (which is selected), "Lỗi tên", "Quản trị viên", "Lịch trực", "Lịch trực", "Danh mục", "Danh mục dịch vụ kỹ thuật", "Danh mục IDC10", and "Danh mục thuốc". The main area is titled "Quản lý người dùng" and contains a search bar with placeholder "Type your keywords here" and a magnifying glass icon. Below the search bar is a blue button labeled "Thêm người dùng". The main content area displays a table of patient information:

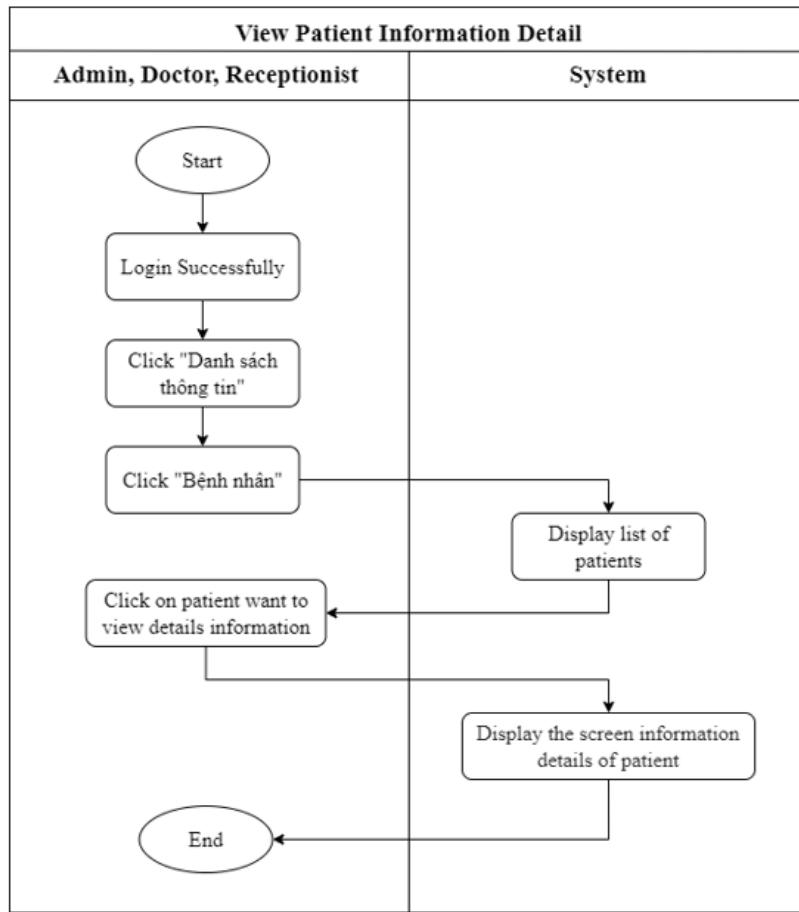
#	Mã Bệnh nhân	Tên	Email	Điện thoại	Ngày sinh	Giới tính	
1	75	Bùi Trí	tribm14@gmail.com	0385617414	02/03/1999	Nam	
2	72	Việt Hoàng	vieth12@gmail.com	0385617412	09/03/2001	Nam	
3	76	Cao Duy	duycao15@gmail.com	0385617415	11/12/1996	Nam	
4	77	Trần Thúy	thuytran16@gmail.com	0385617416	04/06/1986	Nữ	
5	91	Lê Tuấn Anh		0355558474	10/01/1997	Nam	
6	92	Hoàng Linh	hoaqlinh1@gmail.com	0369782088	01/01/1999	Nam	
7	93	Hoàng Việt	viethqhd@gmail.com	0395558546	01/01/1999	Nam	
8	1	System Admin	admin@gmail.com	0123456789	12/01/1999	Nam	
9	96	Bùi Hữu Minh Trí		0352524027	01/01/1999	Nam	

Functional Requirement:

- Admin, doctor can view list of patients.

3.7.2 Patient Information Detail

The following figure show how this function is triggered:



Screen layout:

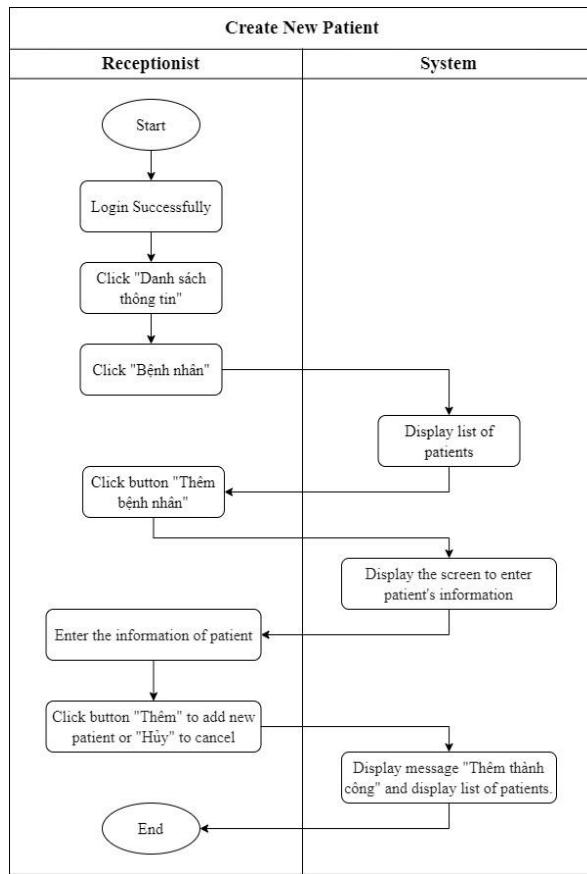
The screenshot shows the 'Quản lý người dùng' (User Management) screen. On the left, there's a sidebar with navigation links like 'Danh sách thông tin', 'Bác sĩ', 'Bệnh nhân', etc. The main area has a table listing users with columns: #, Mã bệnh nhân, and Tên. A modal window is open over the table, containing fields for: Điện thoại (*), Họ và tên (*), Giới tính (*), Ngày sinh (*), Email address, and Address. The 'Lưu' (Save) button is visible at the bottom right of the modal. To the right of the modal, there's a list of users with columns: Điện thoại, Ngày sinh, and Giới tính, each with edit and delete icons.

Function requirement:

- Admin, Doctor, Receptionist can view information of patient detail.

3.7.3 Create New Patient Information

The following figure show how this function is triggered:



Screen layout:

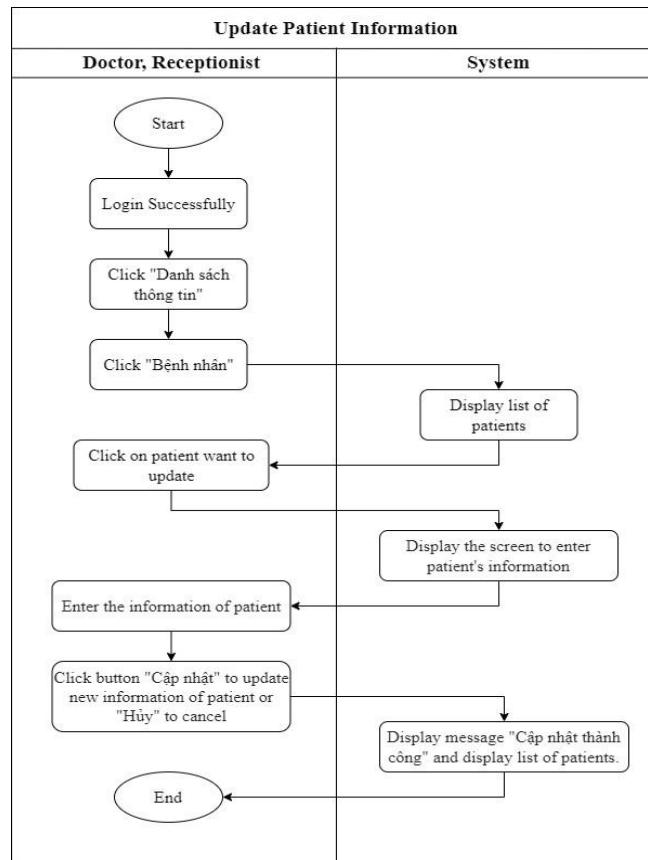
The screenshot shows the 'Quản lý người dùng' (User Management) screen. On the left, there is a sidebar with navigation links: 'Danh sách thông tin', 'Bác sĩ', 'Bệnh nhân', 'Lễ tân', 'Quản trị viên', 'Lịch trực', 'Lịch trực', 'Danh mục', 'Danh mục dịch vụ kĩ thuật', 'Danh mục bệnh ICD10', and 'Danh mục thuốc'. The main area displays a table of patients with columns: #, Mã bệnh nhân, Tên, Điện thoại, Ngày sinh, Giới tính, and actions (Edit, Delete). A modal dialog is open, titled 'Thêm', for adding a new patient. It contains fields for 'Điện thoại (*)', 'Họ và tên (*)', 'Giới tính (*)', 'Ngày sinh (*)', 'Email address', and 'Address'. At the bottom of the dialog are 'Hủy' and 'Lưu' buttons.

Function requirement:

- Receptionists add new patients.

3.7.4 Update Patient Information

The following figure show how this function is triggered:



Screen layout:

The screenshot shows a user management interface. On the left, there's a sidebar with navigation links like 'Danh sách thông tin', 'Bác sĩ', 'Bệnh nhân', etc. The main area is titled 'Quản lý người dùng'. A modal window is open for updating patient information. The modal fields include:

- Điện thoại (*) : 0369782088
- Họ và tên (*) : Hoàng Linh
- Giới tính (*) : Nam
- Ngày sinh (*) : 01/01/1999
- Email address : hoanglinh1@gmail.com
- Address : 40 - Dang Nguyen Can

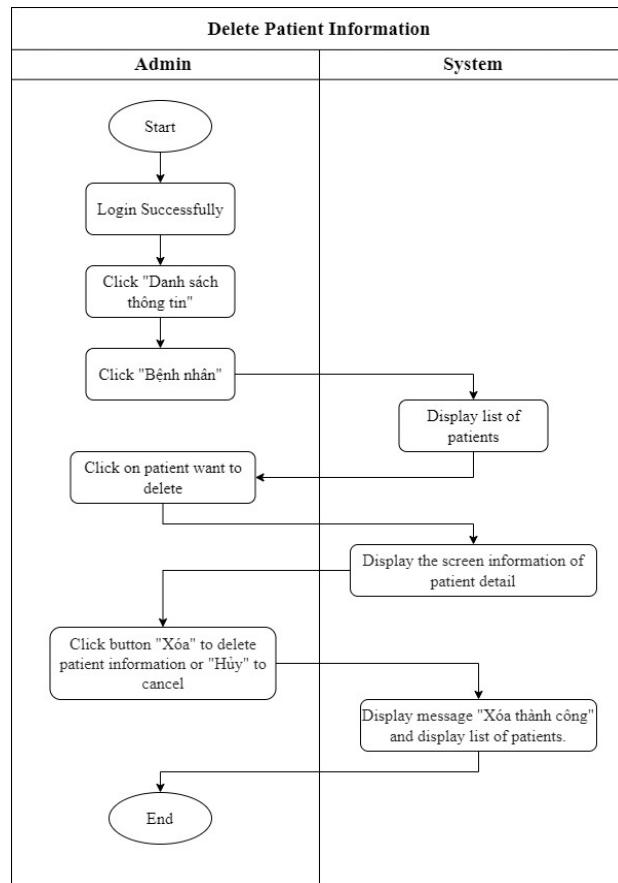
 At the bottom of the modal are 'Hủy' and 'Lưu' buttons. To the right of the modal, a table lists patient records with columns: Mã bệnh nhân, Tên, Điện thoại, Ngày sinh, Giới tính. Each row has edit and delete icons. The table shows 6 rows of data, with page navigation at the bottom indicating 'Showing 1 to 6 of 41 entries'.

Function requirement:

- Doctor, receptionists can update new information about patients.

3.7.5 Delete Patient Information

The following figure show how this function is triggered:



Screen layout:

The screenshot shows a user management interface. On the left, there's a sidebar with navigation links like 'Danh sách thông tin', 'Bác sĩ', 'Bệnh nhân', etc. The main area is titled 'Quản lý người dùng'. A modal window titled 'Delete User' is open, asking 'Bạn có chắc chắn xóa người dùng này không?'. It has two buttons: 'Hủy' (Cancel) and 'Xóa' (Delete). In the background, there's a table listing users with columns: #, Mã bệnh nhân, Tên, Email, Điện thoại, Ngày sinh, Giới tính. The table contains 6 entries. At the bottom, it says 'Showing 1 to 6 of 27 entries'.

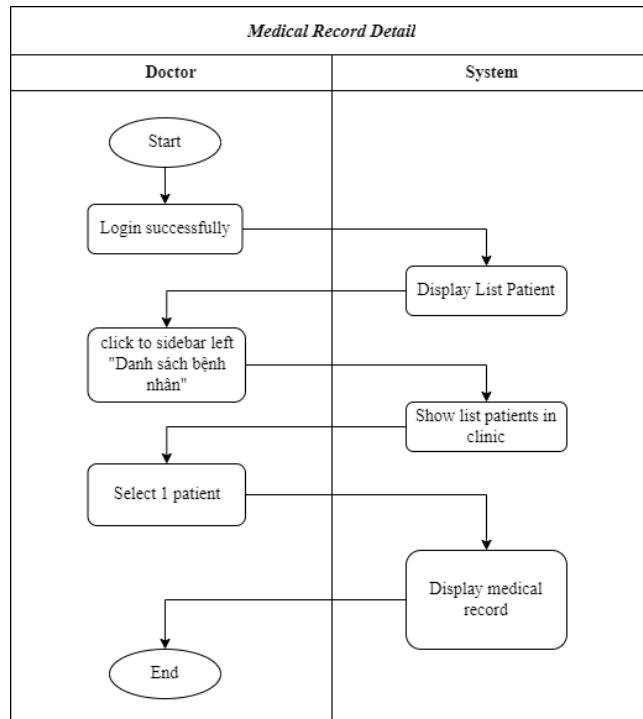
Function requirement:

- Admin can delete information of patients.

3.8 Manage Medical Record

3.8.1 Medical Record Detail

The following figure show how this function is triggered:



Screen layout:

The screenshot shows the CIS application interface. The top navigation bar includes "Linh Nguyen Hoang" and "Đăng xuất". The sidebar on the left has links: "Danh sách bệnh nhân", "Danh sách phác đồ", "Hồ sơ bệnh nhân", and "Lịch trực". The main content area displays patient details:

- Mã BN: 108
- Họ tên: Lưu Duy
- SDT: 0385422617
- Giới tính: Nữ
- Ngày sinh: 12/06/1995
- Địa chỉ: Nam Định

Bệnh án

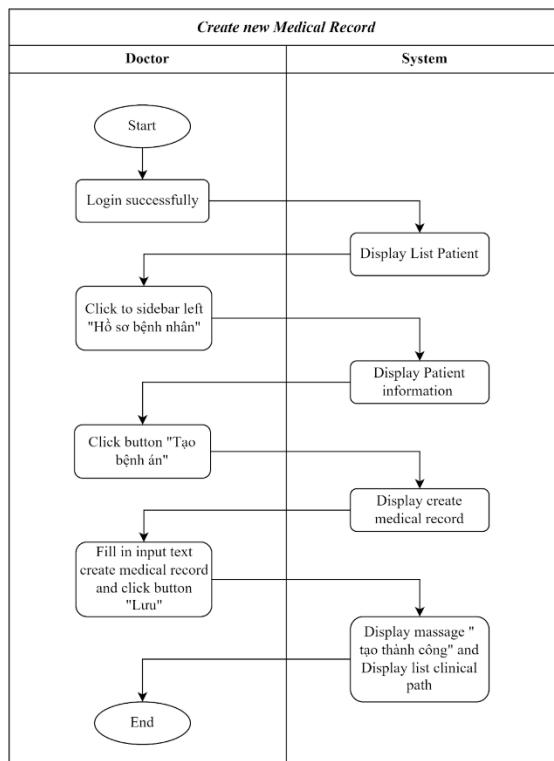
Ngày khám: 25/04/2022
Kết luậ: đau bụng

Functional Requirement:

- Doctor can choose patients to view Medical Record Detail of them.

3.8.2 Create New Medical Record

The following figure show how this function is triggered:



Screen layout:

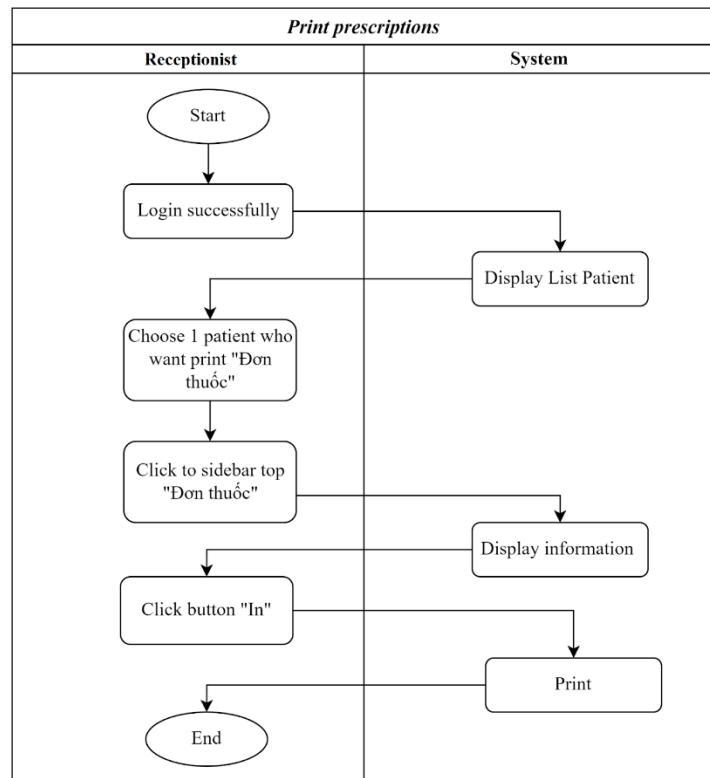
The screenshot shows the CIS application interface. On the left is a dark sidebar with navigation links: 'Danh sách bệnh nhân', 'Danh sách phác đồ', 'Hồ sơ bệnh nhân', and 'Lịch trực'. The main area has a light blue header with tabs: 'Bệnh án' (selected), 'Đơn thuốc', 'Y lệnh', and 'Đơn cam kết'. Below the header, there's a green button labeled 'Khám bệnh' and a yellow button labeled 'Hủy khám bệnh'. A red button labeled 'Kết thúc' and a blue button labeled 'Lưu thông tin' are also visible. The main content area displays patient information: 'Mã BN: 112', 'Họ tên: Nguyễn Hữu Huy', 'Số điện thoại: 0355818489', 'Giới tính: Nam', 'Ngày sinh: 01/01/1999', 'Địa chỉ: hai duong', 'Nhiệt độ: 36 (°C)', 'Chiều cao: 169 (cm)', 'Huyết áp: 40 (mmHg)', 'Cân nặng: 75 (kg)', 'Triệu chứng: đau tai', 'Chẩn đoán đầu: Sỏi biến chứng viêm tai giữa H67.1', 'Chẩn đoán cuối: Bệnh phổi thương hàn A', 'Bệnh kèm theo: Bệnh tè, không đặc hiệu. Sốt phát ban dịch tê do ch�y rận Rickettsia prowazekii.', and 'Kết luận: viêm tai'.

Functional Requirement:

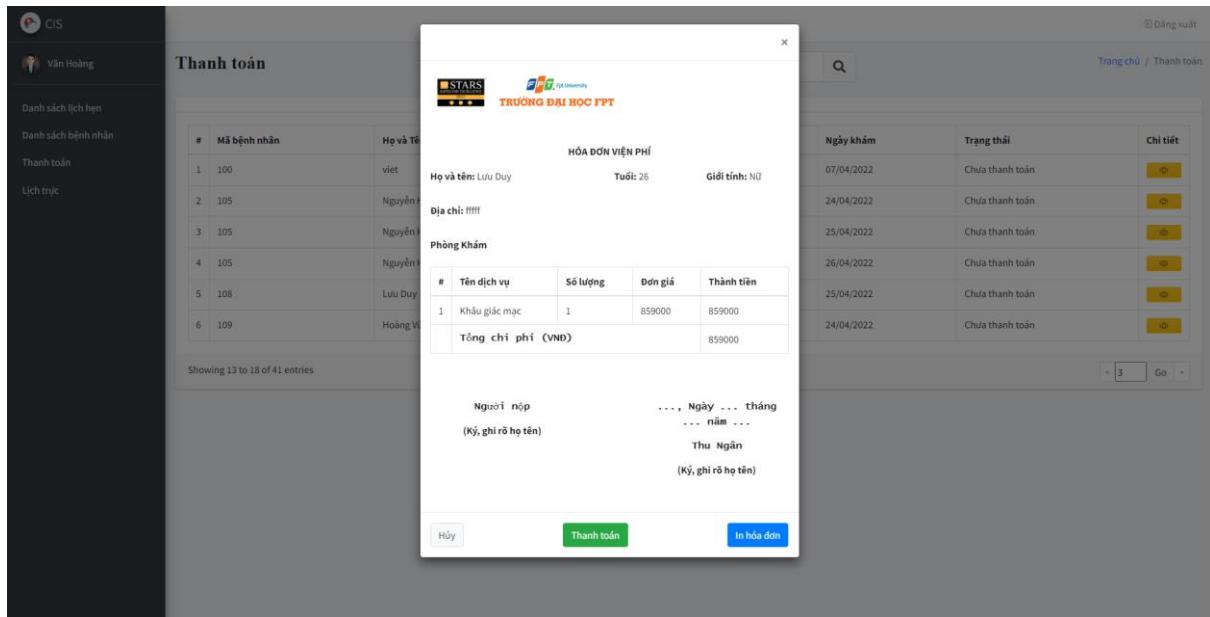
- Doctor can Create Medical Record Detail of patients.

3.8.3 Print prescriptions

The following figure show how this function is triggered:



Screen layout:



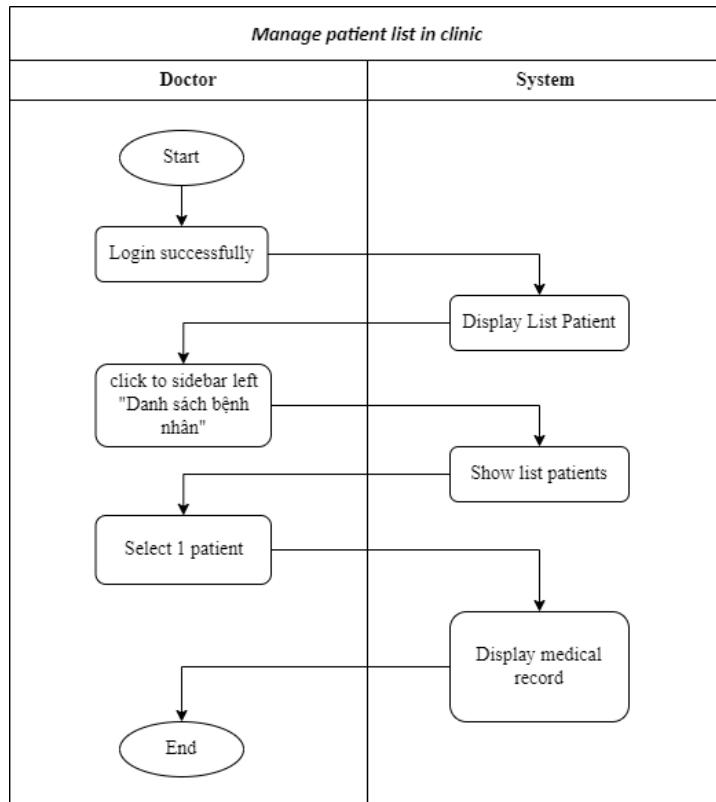
Functional Requirement:

- Doctor can choose a patients and Print prescriptions of them.

3.9 Manage Clinic

3.9.1 Manage patient list in clinic

The following figure show how this function is triggered:



Screen layout:

The screenshot shows the CIS application interface with the following panels:

- Danh sách bệnh nhân đang khám:** A table listing patients currently being examined. It includes columns for STT, Tên (Name), and Độ ưu tiên (Priority). Two patients are listed: Vũ Văn Quân and Nguyễn Hữu Huy. Each patient has three buttons: Xem (View), Chờ khám (Wait for examination), and Đã khám (Examined).
- Danh sách bệnh nhân chờ:** A table listing patients waiting for examination. One patient, Nguyễn Văn Dũng, is listed with priority No. Buttons for Xem and Khám (Exam) are shown.
- Dang chờ kết quả xét nghiệm:** A table listing patients whose test results are pending. Two patients, Nguyễn Hoàng Linh Lai LINH and Võ Công Quyền, are listed with priority No. Buttons for Xem and Đã khám are shown.
- Bệnh nhân đã khám trong ngày:** A table listing patients who have been examined today. Two patients, Dương Hào Nam and Họ và Tên, are listed with priority No. Buttons for Xem are shown.

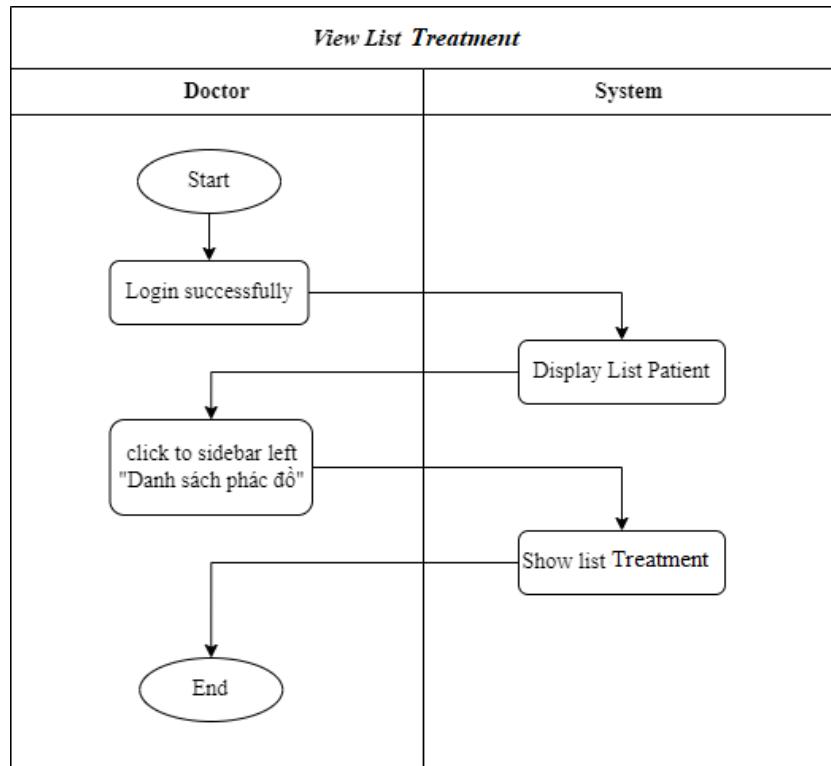
Functional Requirement:

- Doctor and Receptionist view patient list in clinic.

3.10 Manage Treatment

3.10.1 View List Treatment

The following figure show how this function is triggered:



Screen layout:

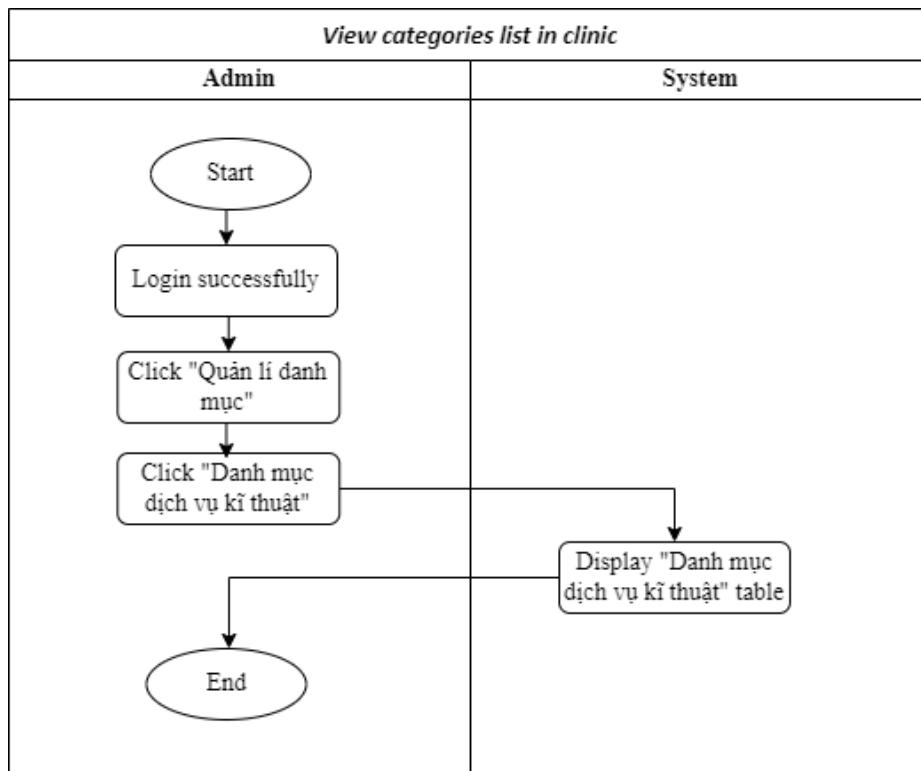
The screenshot shows the 'Treatment List' screen of the CIS application. The top navigation bar includes 'Tạo mới', 'Tim kiếm', 'Đăng xuất', and a user profile for 'Linh Nguyen Hoang'. On the left, there is a sidebar with links: 'Danh sách bệnh nhân', 'Danh sách phác đồ', 'Hồ sơ bệnh nhân', and 'Lịch trực'. The main content area has search filters: 'Phân loại' (Thuốc selected), 'Tên chuyên khoa' (dropdown), and 'Tên phác đồ' (text input). Below these are two tables. The first table lists treatments by specialty: 'Chuyên khoa' (Mắt, Tai mũi họng) and 'Tên phác đồ' (Điều trị mắt, Điều trị tai). The second table lists treatments by name: 'Tên phác đồ' (Điều trị viêm tai, Điều trị mắt) and 'Trạng thái' (Riêng Tư, Riêng Tư). Each row in the second table includes a 'Edit' icon. At the bottom, a message says 'Showing 1 to 2 of 2 entries' and there are navigation buttons for page 1 and 'Go'.

Functional Requirement:

- Doctor can view list treatment.

3.11 Manage Category

3.11.1 View List Categories



The following figure show how this function is triggered:

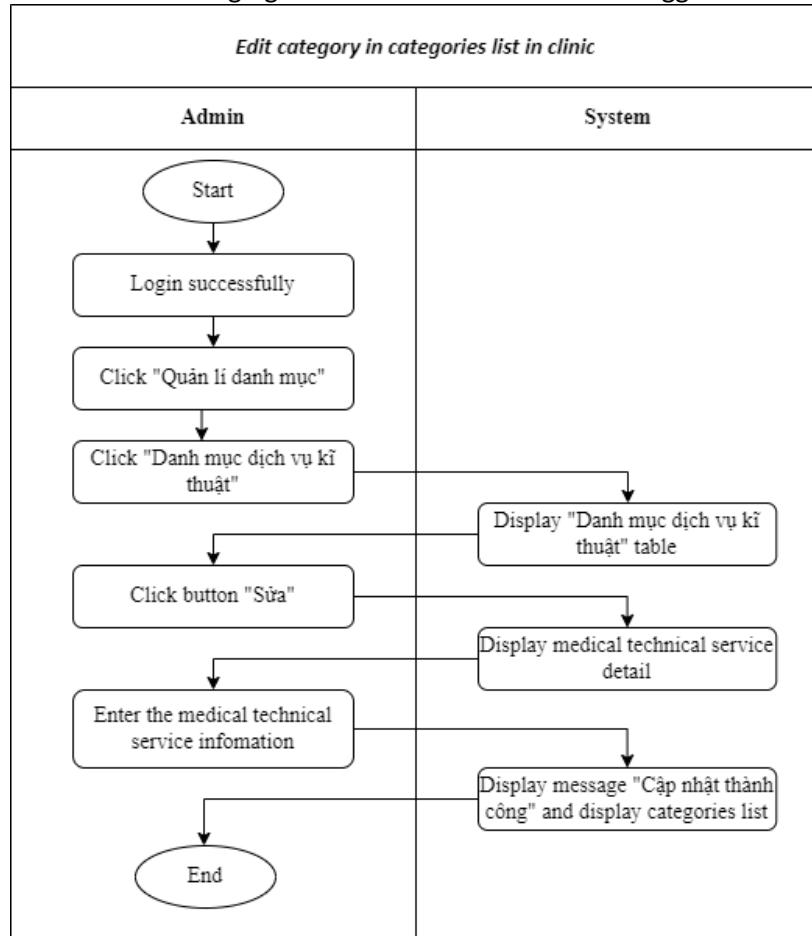
Screen layout:

Functional Requirement:

- Admin can view list categories.

3.11.2 Edit Categories

The following figure show how this function is triggered:



Screen layout:

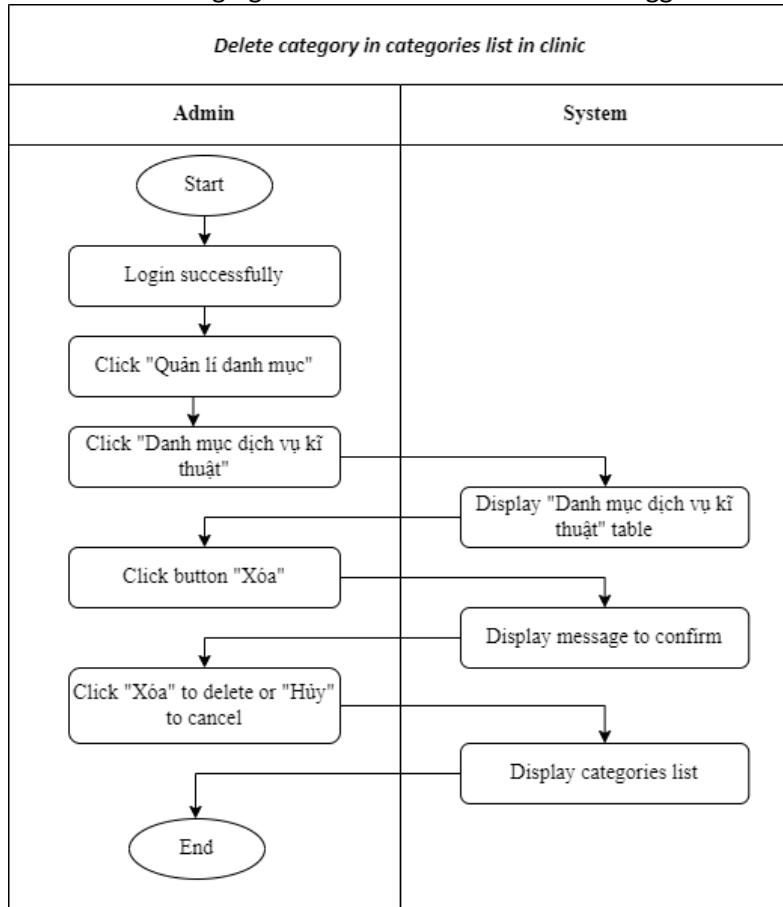
The screenshot shows the 'Quản lý danh mục kỹ thuật' (Manage Category) screen. On the left, there's a sidebar with navigation links like 'Danh sách thông tin', 'Bác sĩ', 'Bệnh nhân', 'Lễ tân', 'Quản trị viên', 'Lịch trực', 'Lịch trực', 'Danh mục', 'Danh mục dịch vụ kỹ thuật', 'Danh mục bệnh ICD10', and 'Danh mục thuốc'. The main area has a table titled 'Table' with columns: #, Chuyên khoa, Tên dịch vụ, and Mô tương đương. There are 6 rows of data. A modal dialog is open over the table, titled 'Chuyên khoa'. It contains fields: 'Hồi sức cấp cứu và Chống độc' (selected in a dropdown), 'Tên dịch vụ' (Thiết khí nhân tạo không xâm nhập phương thức CPAP [giờ theo tết]), 'Code' (01.0130.0209), and a 'Lưu' (Save) button. At the bottom of the modal, it says 'Showing 1 to 6 of 2705 entries'.

Functional Requirement:

- Admin can edit and update list categories.

3.11.3 Delete Categories

The following figure show how this function is triggered:



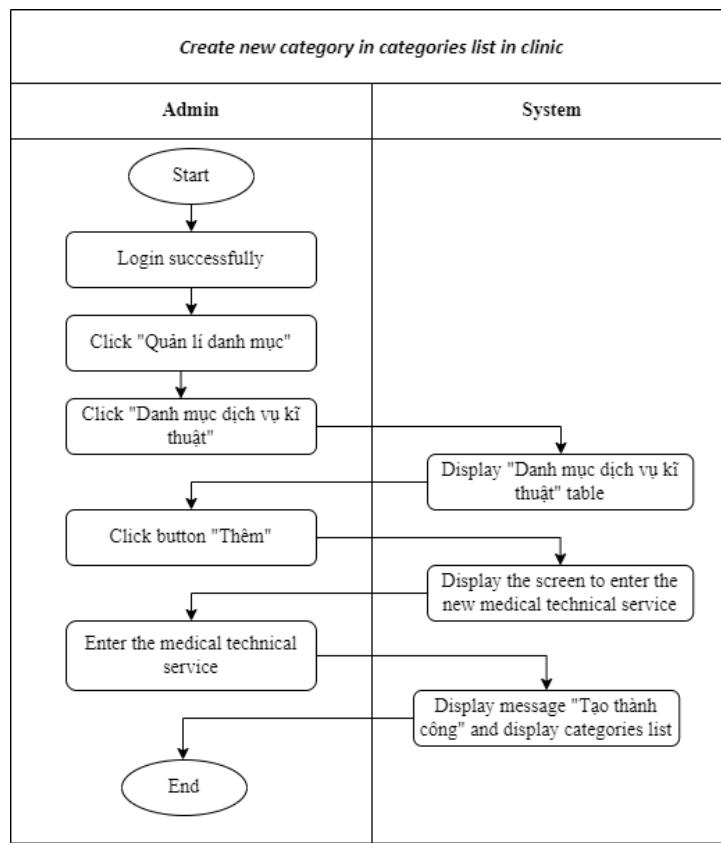
Screen layout:

The screenshot shows a web-based application interface for managing categories. On the left, there's a sidebar with navigation links like 'Danh sách thông tin', 'Bác sĩ', 'Bệnh nhân', etc. The main area has a title 'Quản lý danh mục kỹ thuật'. Below it is a table with columns: #, Chuyên khoa, Tên dịch vụ, and Mã tương đương. The table contains 6 rows of data. A modal dialog box is open in the center, titled 'Xóa danh mục', asking 'Bạn có chắc chắn xóa danh mục này?'. It has two buttons: 'Hủy' (Cancel) and 'Xóa' (Delete). At the bottom of the page, there's a footer with 'Showing 1 to 6 of 2705 entries' and a 'Go' button.

Functional Requirement:

- Admin can delete list categories.

3.11.4 Create New Categories



The following figure show how this function is triggered:

Screen layout:

The screenshot shows the 'Quản lý danh mục kỹ thuật' (Manage medical technical service categories) screen. On the left, there's a sidebar with navigation links like 'Danh sách thông tin', 'Bác sĩ', 'Bệnh nhân', 'Lê tần', 'Quản trị viên', 'Lịch trực', 'Lịch trực', 'Danh mục', 'Danh mục dịch vụ kỹ thuật', 'Danh mục bệnh ICD10', and 'Danh mục thuốc'. The main area has a table titled 'Quản lý danh mục kỹ thuật' showing six entries. A modal window titled 'Chuyên khoa' is open, prompting the user to enter a service name ('Tên dịch vụ') and code ('Code'). The table columns include '#', 'Chuyên khoa', 'Mã tương đương', and actions represented by icons. At the bottom of the table, it says 'Showing 1 to 6 of 2705 entries'.

Functional Requirement:

- Admin can create list categories.
- Validate information when create new categories.

4. Non-Functional Requirements

4.1 External Interfaces

4.1.1 User Interfaces

- UI-1: The websites will have a user-friendly interface and are easy to use.
- UI-2: The layout of the screen is streamlined, easy to operate, and implement for the user.
- UI-3: Buttons have distinctive features and colours but are in sync with the interface.
- UI-4: When the user is on a screen or function, the position of that function displayed on the sidebar will be bright and synchronized with the interface.
- UI-5: Notifications displayed on the screen have a friendly banner outlining the content for the user.
- UI-6: The screen displays shall conform to the Process Impact Internet Application User Interface Standard, Version 2.0.
- UI-7: The system will provide a website help link that is displayed explaining how to use that page to the user.
- UI-8: Web pages will allow users to copy and paste using a mouse, in addition to using a mouse and keyboard combination.

4.1.2 Software Interfaces

- SI-1: Login System
 - SI-1.1: The CIS will pass the phone number and password of the user to the sign-in the system through a programming interface.
 - SI-1.2: The sign-in system will notify CIS that the user exists or not.
- SI-2: Manage System: The CIS shall communicate with the Manage System through a programmatic interface for the following operations:
 - SI-2.1: To allow an Admin to view all Doctor, Receptionist, Schedule, Categories.
 - SI-2.2: To allow a Receptionist to view all Patient, Schedule.
 - SI-2.3: To allow a Doctor to view all Patient, Medical Record.

4.1.3 Hardware Interfaces

- No hardware interfaces have been identified.

4.1.4 Communications Interfaces

- CI-1: The CIS system shall send an email to patient when make an appointment successful.

- CI-2: The CIS system shall send information about appointment from patient to receptionist.

4.2 Quality Attributes

4.2.1 Usability

- The languages are Vietnamese.
- The user interface of the solution should be elegant, easy to navigate, and user-friendly
- All the titles or names of functions should be easy to understand and reflect the purpose of the function.
- All error message should be clear for all users
- All related functions for each type of user should be grouped into categories and placed on a navigation bar for the efficiency of interactions.
- User should be able to learn to use the specified functions in no time
- Links, buttons are easily clickable.
- Links are easily recognizable, which looks clickable.
- Main functions are organized into tabs for easier access.
- Links, buttons, and checkboxes are easily clickable.
- The Front-end web application should support Chrome and Firefox browsers.

4.2.2 Reliability

The system runs 24/7.

- The average time it takes to repair the system if a module fails is about 6-12 hours.
- The system is only allowed to stop working for at most 12 hours since the problem occurred so that users can continue using the system.

4.2.3 Performance

- Response time as fast as possible.
- There will be no restriction in time or place for the use of the software built from the specifications produced under this contract.

4.2.4 Dependability

4.2.4.1 Security

- Allow the separation between front-end and back-end sides. The front-end can be developed as standalone client applications (web, mobile, etc.), while the back-end provides APIs for client consumption.

- All passwords must be hashed using the PasswordEncoder.

4.2.4.2 Safety

- This product is a web application running on the web browser so it will not affect any other application or user's hardware, which may lead to data loss or system damage. The software is designed mainly for devices with an appropriate wide display so for maintaining the correctness of the user's actions and data, the user should use this application on a device with a wide enough screen so the software can fit the screen.

4.2.5 Supportability

- The system will have maintenance periods to improve user security.
- The system will update content updates for users.
- The system has counsellors to support mentees and mentors if they have any problems.

4.2.6 Design Constraints

- End-user's Environment: Windows.
- Support languages: Vietnamese.
- Web websites must be responsive and snappy.
- The user must have a stable connection to the internet.

4.2.7 Support Documents

- F-CIS User guides

5. Requirement Appendix

5.1 Business Rules

ID	Rule Definition
BR-01	Medical examination time for one patient is 15 minutes, at same time, 1 doctor can't book once more time.
BR-02	Appointment must be between 8:00 A.M. and 5:00 P.M. local time, inclusive.
BR-03	Can't make an appointment with doctor when their status is busy.
BR-04	All bills need to be paid before leaving.
BR-05	Only Admin can manage all category.
BR-06	Only doctor can update, edit, or delete Medical Records, Clinical path.
BR-07	Only admin can register for doctor and receptionist.
BR-08	Network transmissions that involve financial information or personally identifiable information require 256-bit encryption.
BR-09	A receptionist can make an appointment for a patient when they can't make an appointment for other reasons.

5.2 Common Requirements

N/A

5.3 Application Messages List

#	Message code	Message Type	Context	Content
1	MSG01	In line	There is not any search result	<i>No search results.</i>
2	MSG02	In red, under the text box	Input-required fields are empty	<i>The * field is required.</i>
3	MSG03	Toast message	Updating asset(s) information successfully	<i>Update asset(s) successfully.</i>
4	MSG04	Toast message	Adding new asset successfully	<i>Add asset successfully.</i>
5	MSG05	Toast message	Confirming email of asset hand-over is sent successfully	<i>A confirmation email has been sent to (email address).</i>
6	MSG06	Toast message	Resetting asset information successfully	<i>Return asset(s) successfully.</i>
7	MSG07	Toast message	Deleting asset information successfully	<i>Delete asset(s) successfully.</i>
8	MSG08	In red, under the text box	Input value length > max length	<i>Exceed max length of (max_length).</i>
9	MSG09	In line	Username or password is not correct when clicking sign-in	<i>Incorrect user name or password. Please check again.</i>

5.4 Other Requirements

N/A

IV. Software Design Description

1. System Design

1.1 System Architecture

1.1.1 Overall Architecture

1.1.1.1 System Architectural Diagram

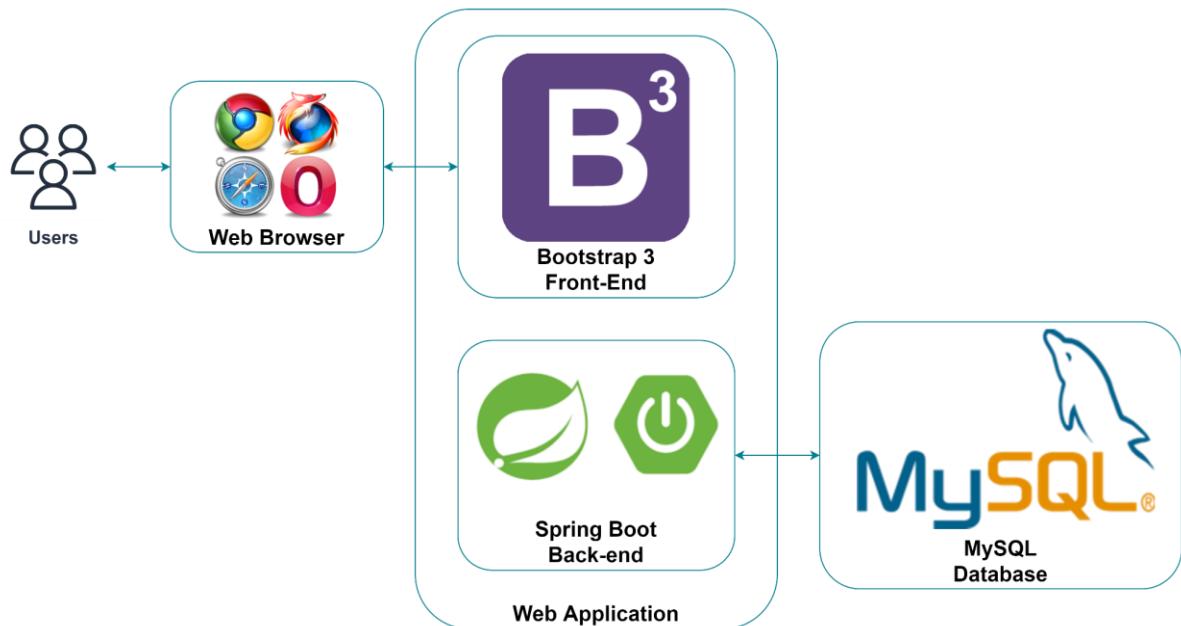


Figure: Overall system architectural diagram

1.1.1.2 Component Explanation

1.1.1.2.1 Front-end of Web Application

In the FCIS system, front-end of web application is responsible for interacting with users through UI in the web site. It displays data received from back-end. After that, it receives information from user and sends data to back-end component.

1.1.1.2.2 Back-end of Web Application

In the FCIS system, back-end of web application is responsible for receiving data from front-end, processing it, updating data into the database and sending response back.

1.1.1.3 Technology and Framework Explanation

1.1.1.3.1 Bootstrap 3



Bootstrap 3 is the version that has been supported by default Responsive, and for bootstrap 2.xx versions, developers will have to add bootstrap-responsive.css if they want to use responsive. And in bootstrap version 3 the font glyphicon-halflings-regular will be replaced for glyphicon-halflings-png. If you are not satisfied with the current interface color. And bootstrap 3 will give you a sharper interface by adding the link style bootstrap.theme.css (or bootstrap-theme.min.css) after the link style bootstrap.css (or bootstrap.min.css).

1.1.1.3.2. Spring Boot



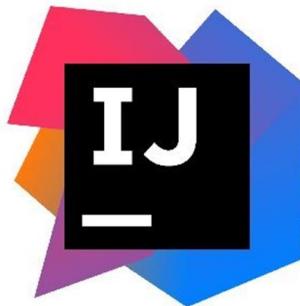
Spring is a framework that provides a comprehensive infrastructure to help programmers develop Java applications. Spring Boot is an extension of Spring Framework that eliminates the complicated configuration steps that Spring requires. We can create an application using Spring Boot and run it in just a few minutes. So, we use it to develop the service system in the back-end component.

1.1.1.3.3. MySQL



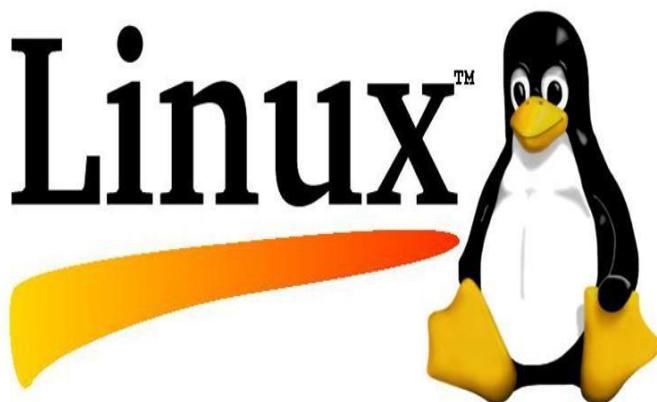
MySQL is an open-source relational database management system (RDBMS). A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

1.1.1.3.4. IntelliJ IDEA



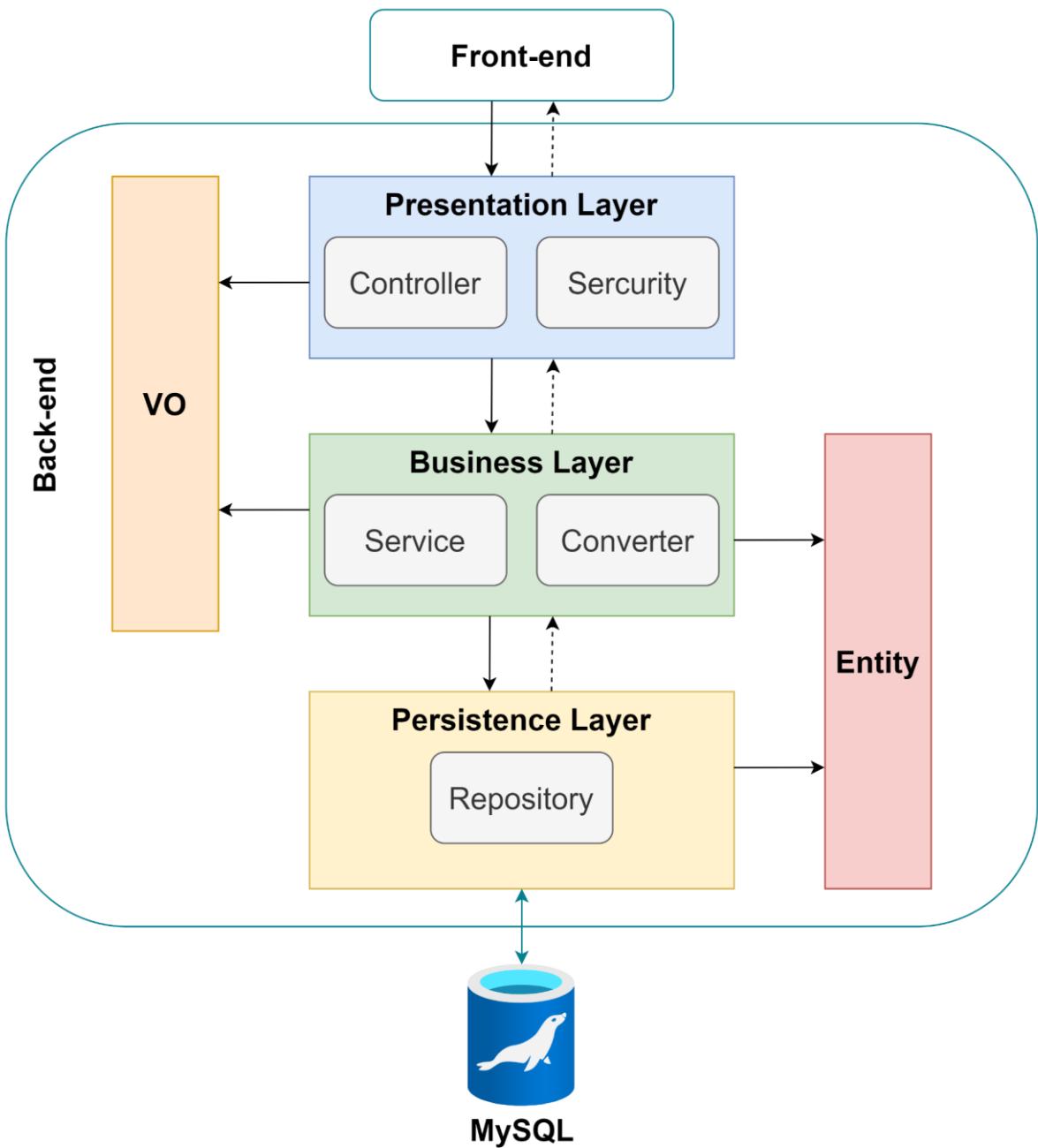
IntelliJ IDEA is an integrated development environment (IDE) written in Java for developing computer software. It is developed by JetBrains (formerly known as IntelliJ), and is available as an Apache 2 Licensed community edition, and in a proprietary commercial edition. Both can be used for commercial development.

1.1.1.3.5. Linux



Linux® is an open source operating system (OS). An operating system is the software that directly manages a system's hardware and resources, like CPU, memory, and storage. The OS sits between applications and hardware and makes the connections between all of your software and the physical resources that do the work.

1.1.2 F-CIS Architecture



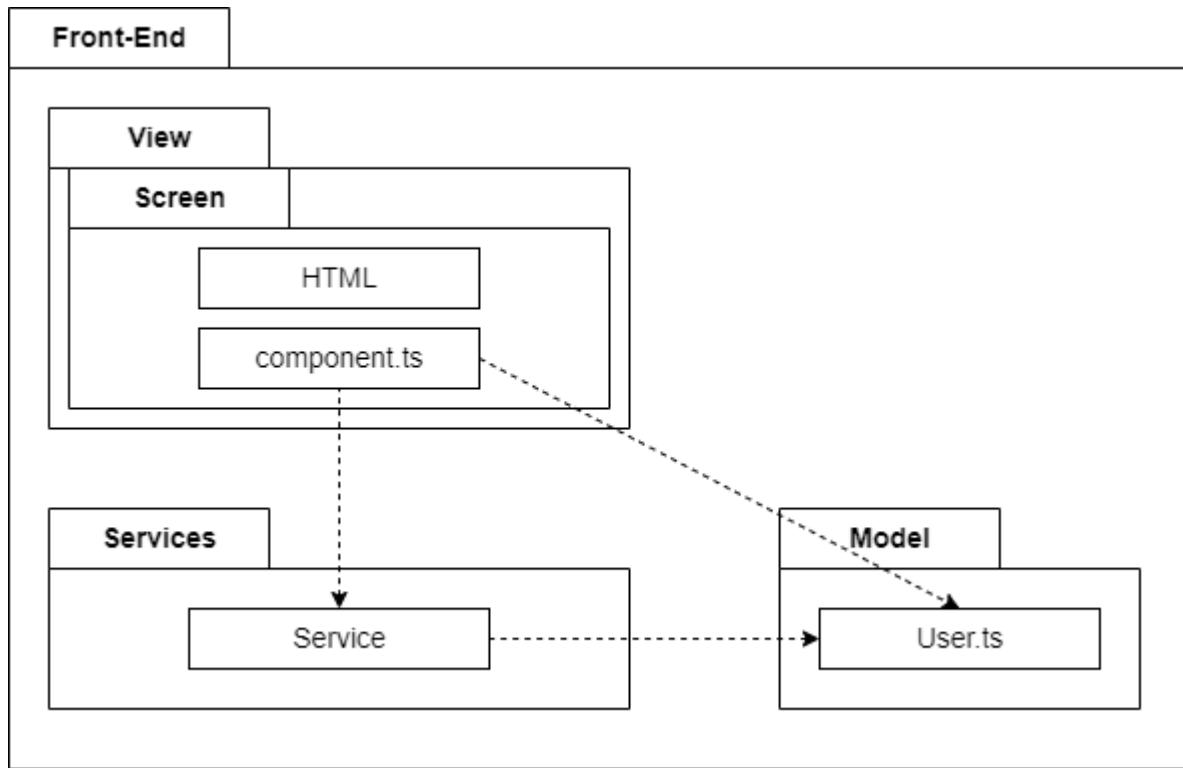
No	Layer/Element	Description
1	Presentation Layer	<ul style="list-style-type: none"> - Include Controller and Security classes - The presentation layer handles all HTTP requests, translates JSON parameters to objects and back, and handles authentication. It does not handle any of the business logic, it

		simply handles request, authenticates it and passes it to the business layer.
2	Business Layer	<ul style="list-style-type: none"> - Include Service classes - The business layer handles all the business logic. Validation and authorization are also done in this layer.
3	Persistence Layer	<ul style="list-style-type: none"> - Include Repository classes - This layer contains all the storage logic (database queries), and translates business objects from and to database rows.
4	Entity	<ul style="list-style-type: none"> - Include Entity classes - This element contains all the classes that map to tables/collections in databases.
5	VO	<ul style="list-style-type: none"> - Include VO classes - This element includes classes that encapsulate data for transmission between front-back-end or between Presentation Layer and Business Layer. The creation of a VO is to reduce the amount of unnecessary information that needs to be transferred, while also increasing security.

1.2 Package Diagram

1.2.1 F- CIS Front-end

1.2.1.1 Diagram

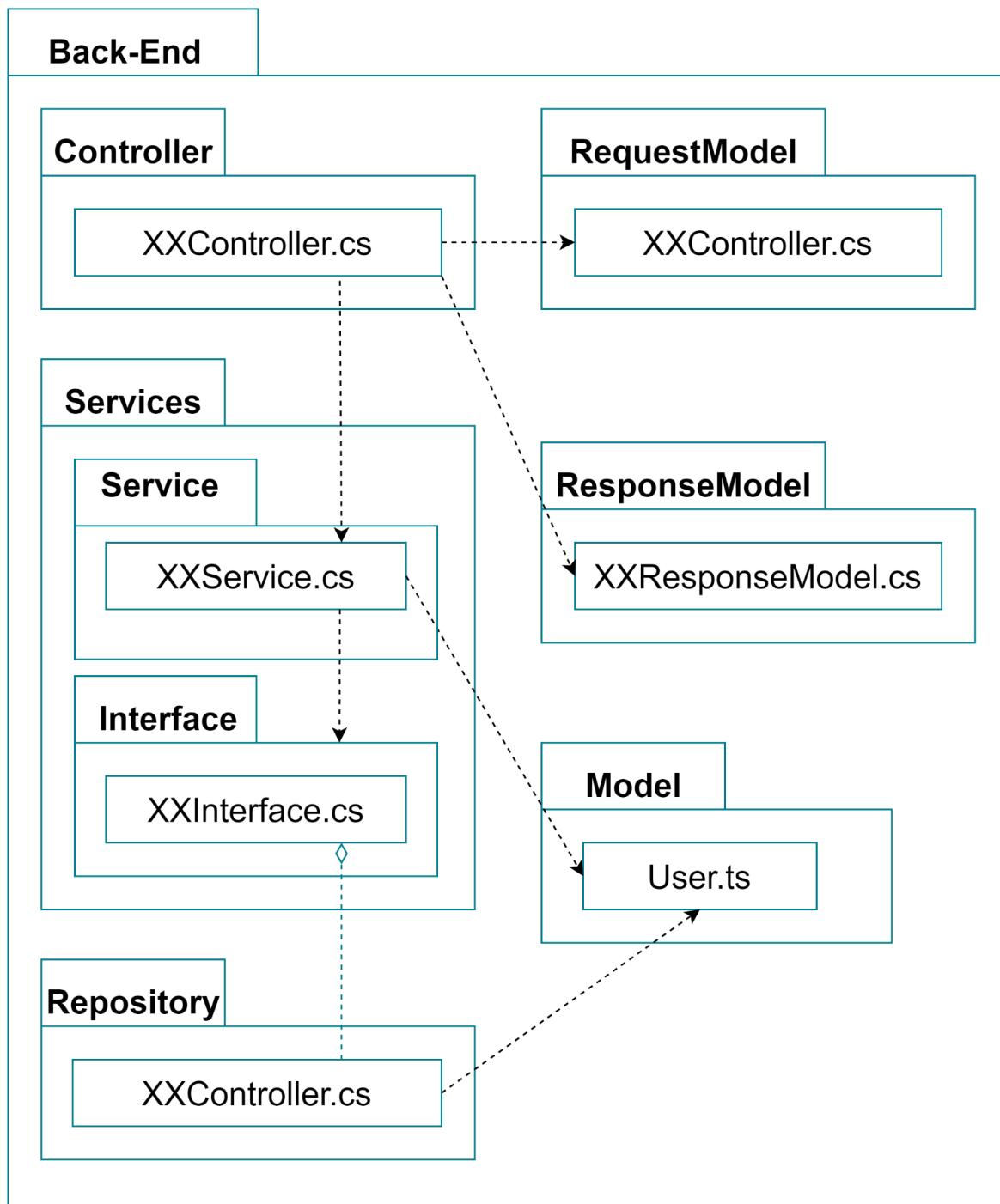


1.2.1.2 Package Descriptions

No	Package	Description
01	Views	Views The layout common folder contains shared interface parts
02	Servies	The services folder contains public api and storage token
03	Model	The model folder contains declare properties to get results from API

1.2.2 F- CIS Back-end

1.2.2.1 Diagram



1.2.2.2 Package Descriptions

No	Package	Description
01	RequestModel	The common folder contains all common classes
02	ViewModel	The data folder contains migrations and DbContext
03	Controller	The controller folder contains public api and business logic
04	Models	The model folder contains the core entity
05	Services	The service folder integrates third-party services
06	Repositories	The repository folder contains classes to communicate with the database

2. Database Design

2.1. Class Specifications

2.1.1 CIS Back-end

2.1.1.1 Controller

2.1.1.1.1 UserRestController

No	Method	Description
1	getData	Using it to show all accounts of users.
2	saveUser	Using it to save after creating and editing accounts of users.
3	deleteUser	Using it to delete users.

4	getDataById	Using it to find users by id.
5	getLoggedUserName	Using it to get information about user name.
6	getDataUserLogin	Using it to get data about user login.
7	getAllDoctor	Using it to show all users is the role of a doctor.

2.1.1.1.2 Catalog Controller

2.1.1.1.2.1 CatalogIcdRestController

No	Method	Description
1	getDataICD	To display all disease lists ICD-10.
2	getDataById	Using it to find ICD-10 by id.
3	saveICD	Using it to save after creating a new disease ICD-10 and editing ICD-10.
5	deleteICD	Using it to delete disease ICD-10.

2.1.1.1.2.2 CatalogMedicineRestController

No	Method	Description
1	getDataMedicine	To display all medicine.
2	getDataMedicineById	Using it to find medicine by id.

3	addEditMedicine	Using it to save after creating a new medicine and editing medicine.
5	deleteMedicine	Using it to delete medicine.

2.1.1.1.2.3 CatalogServiceRestController

No	Method	Description
1	getData	To display all services.
3	addEditService	Using it to save after creating a new service and editing service.
5	deleteService	Using it to delete service.

2.1.1.1.3 App Controller (Web)

2.1.1.1.3.1 AdmissionRecordRestController

No	Method	Description
1	getAdmissionOfDoctor	Using it to display all admission records by doctor current date.
2	updateStatusAdmission OfDoctor	Using it to change the status of a patient.
3	getAllAdmission	Using it to get all patients by doctor current date.
4	getBookingById	Using it to find data booking by id.
5	getBookingByPhone	Using it to find data booking by phone

6	updateStatusBooking	Using it to change status booking.
7	getRoom	Using it to threading.
8	getCalenderDoctor	Using it to get all the schedules of doctors.
9	addCalenderDoctor	Using it to create a new schedule.
10	deleteCalenderDoctor	Using it to delete a schedule.

2.1.1.1.3.2 BookingRestController

No	Method	Description
1	addAdmission	Using it to get all appointment

2.1.1.1.3.3 MedicineRestController

No	Method	Description
1	getPatientList	Using it to get all patient
2	getDataById	Using it to find medicine by id.

2.1.1.1.3.4 PatientRestController

No	Method	Description
1	getPatientById	Using it to get patient by id

2	updatePatient	Using it to edit a patient
3	getPatientList	Using it to get all patient
4	getPatientRecord	Using it to get a patient record by patient id.
5	getProfilePatientById	Using it to get a profile of a patient by patient id.
6	getPaymentPatientRecord	Using it to get all the payment patient records.
7	getPaymentPatientRecordById	Using it to find payment record by patient id.
8	getPaymentPatientChangeStatus	Using it to get status payment after change.

2.1.1.3.5 PrescriptionOfPatientController

No	Method	Description
1	getPrescriptionOfPatientById	Using it to get prescription of patient by id.
2	updatePrescriptionToPatient	Using it to edit prescription of patient.
3	getPrescriptionByRegimen	Using it to get prescription of patient by regimen.

2.1.1.3.6 RegimenRestController

No	Method	Description
1	getRegimen	Using it to get regimen.

2	getRegimenById	Using it to get regimen by Id.
3	addEditRegimen	Using it to add an edit regimen.
4	getRegimenAllPrescription	Using it to get all prescription of regimen.

2.1.1.3.7 ServiceRestController

No	Method	Description
1	getServiceById	Using it to get services by id.
2	getServiceBySpecializedId	Using it to get service by specialized id.
3	getCriticalPathBySpecializedId	Using it to get critical path by specialized id.
4	getServiceByCriticalPathId	Using it to get service by critical path id.
5	updateServiceToPatient	Using it to update service to patients.
6	getDataSearch	Using it to get all data to search.

2.1.1.3.8 SpecializedRestController

No	Method	Description
1	getPatientList	Using it to get all patients.
2	getAllSpecialized	Using it to get all specialised.

2.1.1.2 Service

-User

2.1.1.2.1 UserService

No	Method	Description
1	save	Using it to save after adding or editing user account information.
2	initAdmin	Using it to set default values.
3	findById	Using it to find users by id.
4	deleteUserById	Using it to delete users by id.
5	getAllUser	Using it to display all users.
6	findByEmail	Using it to find users by email.
7	getLoggedInUser	Using it to get information about user login.
8	findByPhone	Using it to find users by phone.
9	getData	Using to get all data users.
10	getAllDoctor	Using it to get all accounts is the role of a doctor.
11	setUserAuthen	Using to set user authentication.

-Catalog

2.1.1.2.2 ServiceService

No	Method	Description
1	getData	Using to get all data of catalog service.
2	deleteServiceById	Using it to delete service by id.
3	addEditService	Using it to add or edit service.

2.1.1.2.3 MedicineService

No	Method	Description
1	getData	Using it to get all data medicine.
2	deleteMedicineById	Using it to delete medicine by id.
3	addEditMedicine	Using it to add or edit medicine.
4	getDataMedicineById	Using it to get data medicine by id.

2.1.1.2.4 DiseaseListICD10Service

No	Method	Description
1	getData	Using to get all disease ICD-10.
2	getById	Using it to get data icd by id.

3	findById	Using it to find icd by id.
4	deleteIcdById	Using it to delete icd by id.
5	save	Using it to add or edit icd-10.
6	findByName	Using it to find medicine by name.

-MasterData

2.1.1.2.5 MasterDataService

No	Method	Description
1	getMasterDataByName	Using it to get master data by name
2	initData	Using it to initialization master data.
3	delete	Using it to delete master data by id.

2.1.1.2.6 MasterDataConfigService

No	Method	Description
1	findAll	Using it to find all data configure.

-App

2.1.1.2.7 SpecializedService

No	Method	Description

1	getServiceByPatientId	Using it to get service by patient id.
2	getServiceBySpecializedId	Using it to get service by specialized id.
3	getServiceByCriticalPathId	Using it to get service by critical path id.
4	updateServiceToPatient	Using it to update service to patients.
5	getDataByNameAndSpecialized	Using it to get data by name and specialized.

2.1.1.2.8 PrescriptionOfPatientService

No	Method	Description
1	getPrescriptionOfPatientById	Using it to get prescription of patients by id.
2	updatePrescriptionToPatient	Using it to update prescriptions to patients.
3	getPrescriptionByRegimen	Using it to get prescriptions by regimen.

2.1.1.2.9 PatientService

No	Method	Description
1	getPatientById	Using it to get data patients by id.
2	updatePatient	Using it to update patient information.
3	findById	Using it to find patients by id.
4	getPatient	Using it to get all patients from the list.

5	getPatientRecord	Using it to get a patient record by patient id.
6	getProfilePatientById	Using it to get a profile of a patient by patient id.
7	getPaymentPatientRecord	Using it to get all the payment patient records.
8	getPaymentPatientChangeStatus	Using it to get status payment after change.
9	getPaymentPatientRecordById	Using it to find payment record by patient id.

2.1.1.2.10 PatientService

No	Method	Description
1	getDataByName	Using it to find medicine by medicine name.
2	getDataById	Using it to find medicine by medicine id.

2.1.1.2.11 GroupMasterService

No	Method	Description
1	getDataByName	Using it to find group master by name.
2	getAllSpecialized	Using it to get all specialised.

2.1.1.2.11 ExaminationRoomService

No	Method	Description
1	getCalenderDoctor	Using it to get all the schedules of doctors.

2	addCalenderDoctor	Using it to create a new schedule.
3	deleteCalenderDocotr	Using it to delete schedules.

2.1.1.2.11 CMSRegimentService

No	Method	Description
1	getCriticalPathBySpecialized	Using it to get a critical path by specialized.
2	getRegimen	Using it to get all the critical paths.
3	addEditRegimen	Using it to save after add or edit.
4	getRegimenById	Using it to get a critical path by id.
5	getRegimenMedicineById	Using it to get critical path medicine by id.

2.1.1.2.11 AdmissionRecordService

No	Method	Description
1	getAllAdmissionRecordByDoctor CurDate	Using it to get all patients by doctor current date.
2	updateStatusAdmissionOfDoctor	Using it to change the status of a patient.
3	booking	Using it to create new clinical record and create new booking.

4	getBookingById	Using it to find data booking by id.
5	getBookingByPhone	Using it to find data booking by phone.
6	updateStatusBooking	Using it to change status booking.

2.1.2 CIS Front-end

2.1.2.1 Pages

2.1.2.1.1 Admin

No	Method	Description
1	CreateAccount	Create a new account
2	DeleteAccount	Remove an account
3	EditAccount	Edit an account
4	SearchAccount	Search an account
5	ViewAccountList	Display the list of accounts
6	CreateSchedule	Create a new schedule
7	DeleteSchedule	Remove a schedule
8	EditSchedule	Edit a schedule
9	SearchSchedule	Search a schedule

10	ViewScheduleList	Display the list of schedules
11	CreateCategory	Create a category
12	EditCategory	Edit the category
13	DeleteCategory	Remove a category
14	SearchCategory	Search a category
15	ViewCategoryList	Display the list of categories

2.1.2.1.2 Receptionist

No	Method	Description
1	MakeAnAppointment	Make a new appointment
2	ConfirmAnAppointment	Accept or remove an appointment
3	EditAnAppointment	Edit an account
4	SearchAppointment	Search an appointment
5	ViewAppointmentList	Display the list of appointments
6	CreatePatient	Create a new patient
7	DeletePatient	Remove a patient

8	EditPatient	Edit a patient
9	SearchPatient	Search a patient
10	ViewPatientList	Display the list of patients
11	ViewPaymentList	View list of examination bills
12	PrintBill	Print examination bills
13	ViewScheduleList	Display the list of schedules

2.1.2.1.3 Doctor

No	Method	Description
1	ViewPatientList	Display the profile of a patient (examination-ing, examinationed, waiting, priority)
2	CreateTreatment	Create a new treatment regimen
3	DeleteTreatment	Remove a treatment regimen
4	EditTreatment	Edit a treatment regimen
5	SearchTreatment	Search a treatment regimen
6	ViewTreatmentList	Display the list of treatment regimen
7	ViewMedicalRecords	Display the list of patient medical records

8	SearchMedicalRecord	Search a medical record
9	ViewScheduleList	Display the list of schedules

2.1.2.1.4 Patient

No	Method	Description
1	MakeAnAppointment	Make a new appointment

2.1.2.1.5 Common

No	Method	Description
1	Login	Log in to the system
2	Logout	Log out to the system
3	HomePage	Display home page
4	NotFound	Display not found page when wrong path

2.1.2.2 Library

2.1.2.2.1 Common

No	Method	Description
1	stringToDate	Using it to convert string to date.

2	dateToString	Using it to convert date to string.
3	parseDouble	Using it to parse Double.
4	getClaims	Using it to get Claims.
5	validateToken	Using it to validate a token.
6	generateToken	Using it to create a token.
7	generatePassWord	Using it to auto create passwords.

2.1.2.3 Context

2.1.2.3.1 Authenticator

No	Method	Description
1	AdminProvider	Login authentication admin
2	ReceptionistProvider	Login authentication Receptionist
3	DoctorProvider	Login authentication doctor

2.1.2.4 Components

2.1.2.4.1 UI

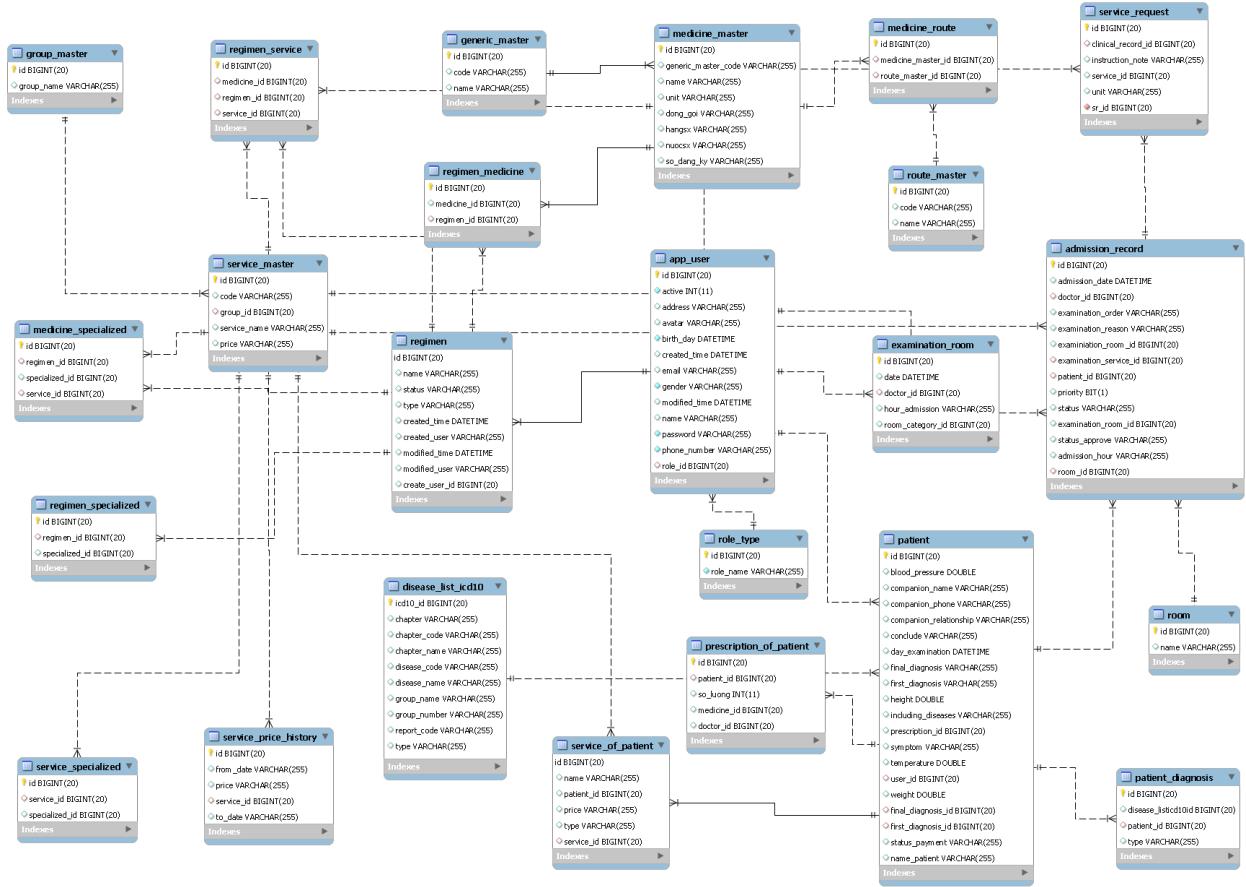
No	Method	Description
1	BackDropLayout	Display the backdrop layout
2	EffectLoading	Display component wrapper of effect loading
3	Notification	Display Notification Modal
4	OptionRadModal	Display Radio Input Modal
5	Loader	Display effect loading
6	SendMsgBox	Display message box
7	YesNoModal	Display option modal

2.1.2.4.2 Layout

No	Method	Description
1	MainLayout	Display main content layout
2	MainNav	Display the navigation
3	Pagination	Display pagination component
4	Login	Display login form

5	Search	Display search box
6	EmptyData	Display empty row
7	THead	Display the head component of table
8	Footer	Display Footer component
9	Header	Display Header component
10	MainLayoutLogged	Display main content layout when logged in
11	MainSlideBar	Display slide bar for doctor and patient
12	TBody	Display the body component of table
13	FilterDate	Display the input date component

2.2. Database Design



2.2.1 App_User

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	active	INT	11	N	N	N	Status of user
3	address	VARCHAR	255	N	N	N	Address of user
4	avatar	VARCHAR		N	N	N	Avatar of user

5	birth_day	DATETIME		N	N	N	Birth of date of user
6	created_time	DATETIME		N	N	N	Time to create of user
7	email	VARCHAR	255	N	N	N	Email of user
8	gender	VARCHAR	255	N	N	N	Gender of user
9	modified_time	DATETIME		N	N	N	Time modified of user
10	name	VARCHAR	255	N	N	N	Name of user
11	password	VARCHAR	255	N	N	N	Password of account user.
12	phone_number	VARCHAR	255	N	N	N	Phone of user account.
13	role_id	VARCHAR	20	N	N	Y	Id of role user.

2.2.2 Admission_record

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	admission_date	DATE		N	Y	N	Date admission of patient
3	doctor_id	BIGINT	20	N	Y	Y	Id of doctor

4	examination_order	VARCHAR	255	N	Y	Y	Oder to examination
5	examination_reason	VARCHAR	255	N	Y	Y	Reason why need examination
6	examination_room_id	BIGINT	20	N	Y	Y	Room to examination
7	examination_service_id	BIGINT	20	N	Y	Y	Service of examination
8	patient_id	BIGINT	20	N	N	Y	Id of patient.
9	priority	BIT	1	N	N	N	Priority of patient.
10	status	VARCHAR	255	N	N	N	Status of patient.
12	status_approve	VARCHAR	255	N	N	N	Status is approved.
13	admission_hour	VARCHAR	255	N	N	N	Hour of admission
14	room_id	BIGINT	20	N	N	Y	Id of room

2.2.3 Examination_room

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id

2	date	DATETIME		N	Y	Y	Working day of doctor
3	doctor_id	BIGINT	20	N	Y	Y	Id of doctor
4	hour_admission	VARCHAR	255	N	Y	N	Hour of admission
5	room_category_id	BIGINT	20	N	Y	Y	Categories of room id

2.2.4 Generic_master

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	code	VARCHAR	255	N	Y	N	Generic code
3	name	VARCHAR	255	N	Y	N	Generic name

2.2.5 Group_master

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	group_name	VARCHAR	255	N	Y	N	Name of group

2.2.6 Medicine_route

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	medicine_master_id	BIGINT	20	N	Y	Y	Id of medicine master
3	route_master_id	BIGINT	20	N	Y	Y	Id of route master

2.2.7 Medicine_master

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	generic_master_code	VARCHAR	255	N	Y	Y	Code of generic
3	name	VARCHAR	255	N	Y	N	Name of medicine
4	unit	VARCHAR	255	N	Y	N	unit of medicine

2.2.8 Medicine_specialized

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id

2	regimen_id	BIGINT	20	N	Y	Y	Id of regimen
3	specialized_id	BIGINT	20	N	Y	Y	Id of specialized
4	service_id	BIGINT	20	N	N	Y	Id of service

2.2.9 Patient

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	blood_pressure	DOUBLE		N	Y	Y	Blood pressure of patient
3	companion_name	VARCHAR	255	N	Y	Y	Patient's companion name
4	companion_phone	VARCHAR	255	N	Y	N	Patient's companion phone
5	companion_relationship	VARCHAR	255	N	Y	Y	Relationship of companion with patient
6	conclude	VARCHAR	255	N	Y	Y	Conclude of doctor
7	day_examination	DATETIME		N	Y	N	Date time to examination

8	final_diagnosis	VARCHAR	255	N	Y	N	Date time final diagnosis
9	first_diagnosis	VARCHAR	255	N	Y	N	Date time first diagnosis
10	height	DOUBLE		N	N	N	Height of patient
11	including_diseases	VARCHAR	255	N	N	N	Including diseases
12	prescription_id	BIGINT	20	N	N	Y	Id of prescription
13	symptom	VARCHAR	255	N	N	N	Symptom of patient
14	temperature	DOUBLE		N	N	N	Temperature of patient
15	user_id	BIGINT	20	N	N	Y	Id of patient
16	weight	DOUBLE		N	N	N	Weight of patient
17	final_diagnosis_id	BIGINT	20	N	N	Y	Id of final diagnosis
18	first_diagnosis_id	BIGINT	20	N	N	Y	Id of first diagnosis
19	status_payment	VARCHAR	255	N	N	N	Status of payment
20	name_patient	VARCHAR	255	N	N	N	Name of patient

2.2.10 Patient_diagnosis

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	disease_listicd10id	BIGINT	20	N	Y	Y	List disease of icd10
3	patient_id	BIGINT	20	N	Y	N	Id of patient
4	type	VARCHAR	255	N	Y	N	Type of diagnosis

2.2.11 Prescription_of_patient

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	patient_id	BIGINT	20	N	Y	Y	Id of patient
3	medicine_id	BIGINT	20	N	Y	N	Id of medicine
4	doctor_id	BIGINT	20	N	Y	N	Id of doctor

2.2.12 Disease_list_icd10

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	icd10_id	BigInt	20	Y	Y	Y	Auto-generation id
2	chapter	VARCHAR	255	N	Y	N	Chapter of ICD10
3	chapter_code	VARCHAR	255	N	Y	N	Code of chapter in ICD10
4	chapter_name	VARCHAR	255	N	Y	N	Name of chapter in ICD10
5	group_number	VARCHAR	255	N	Y	N	Group number in ICD10
6	group_name	VARCHAR	255	N	Y	N	Group name in ICD10
7	type	VARCHAR	255	N	Y	N	Type of Disease in ICD10
8	disease_code	VARCHAR	255	N	Y	N	Code of disease in ICD10
9	disease_name	VARCHAR	255	N	Y	N	Name of disease in ICD10
10	report_code	VARCHAR	255	N	Y	N	Code report in ICD10

2.2.13 Regimen

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	name	VARCHAR	255	N	Y	Y	Name of regimen
3	status	VARCHAR	255	N	Y	N	Status of regimen
4	type	VARCHAR	255	N	Y	N	Type of regimen
5	created_time	DATETIME		N	Y	N	Create time for regimen
6	created_user	VARCHAR	255	N	Y	N	Create regimen for user
7	modified_time	DATETIME		N	Y	N	modified time for regimen
8	modified_user	VARCHAR	255	N	Y	N	modified for regimen
9	create_user_id	BIGINT	20	N	N	Y	Create user by id

2.2.14 Regimen_medicine

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id

2	medicine_id	BIGINT	20	N	Y	N	Id of medicine for regimen
3	regimen_id	BIGINT	20	N	Y	N	Id of regimen for regimen

2.2.15 Regimen_service

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	medicine_id	BIGINT	20	N	Y	Y	Id of medicine for regimen
3	regimen_id	BIGINT	20	N	Y	Y	Id of regimen for regimen
4	service_id	BIGINT	20	N	Y	N	Id of service for regimen

2.2.16 Regimen_specialized

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	regimen_id	BIGINT	20	N	Y	N	Id of regimen
3	specialized_id	BIGINT	20	N	Y	Y	Id of specialized

2.2.17 Role_type

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	role_name	VARCHAR	255	N	Y	N	Name of role

2.2.18 Room

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	name	VARCHAR	255	N	Y	N	Name of room

2.2.19 Route_master

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	code	BIGINT	20	N	Y	Y	Code of route
3	name	VARCHAR	255	N	Y	N	Name of route

2.2.20 Service_master

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	code	VARCHAR	255	N	Y	N	Code of service
3	group_id	BIGINT	20	N	N	Y	Id of group for service
4	service_name	VARCHAR	255	N	N	N	Name of service

2.2.21 Service_of_patient

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	name	VARCHAR	255	N	Y	N	Name of service of patient
3	patient_id	BIGINT	20	N	N	Y	Id of patient
4	price	VARCHAR	255	N	N	N	Price of service
5	type	VARCHAR	255	N	N	N	Type of service
6	service_id	BIGINT	20	N	N	Y	Id of service

2.2.22. Service_price_history

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	from_date	VARCHAR	255	N	Y	N	Date time of created
3	price	VARCHAR	255	N	N	N	Price of service
4	service_id	BIGINT	20	N	N	Y	Id of the service
5	to_date	VARCHAR	255	N	N	N	Date time of created

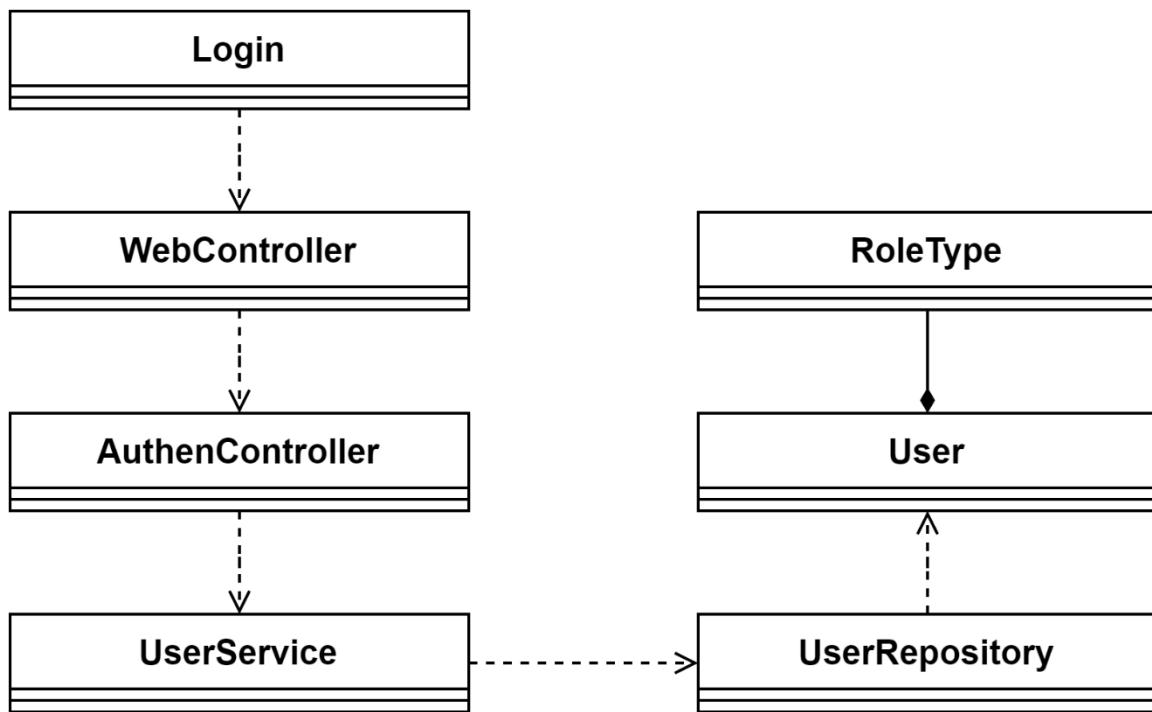
2.2.23. Service_specialized

#	Field name	Type	Size	Unique	Not Null	PK/FK	Notes
1	id	BIGINT	20	Y	Y	Y	Auto-generation id
2	service_id	BIGINT	20	N	N	Y	Id of service.
3	specialized_id	BIGINT	20	N	N	Y	Id of specialised

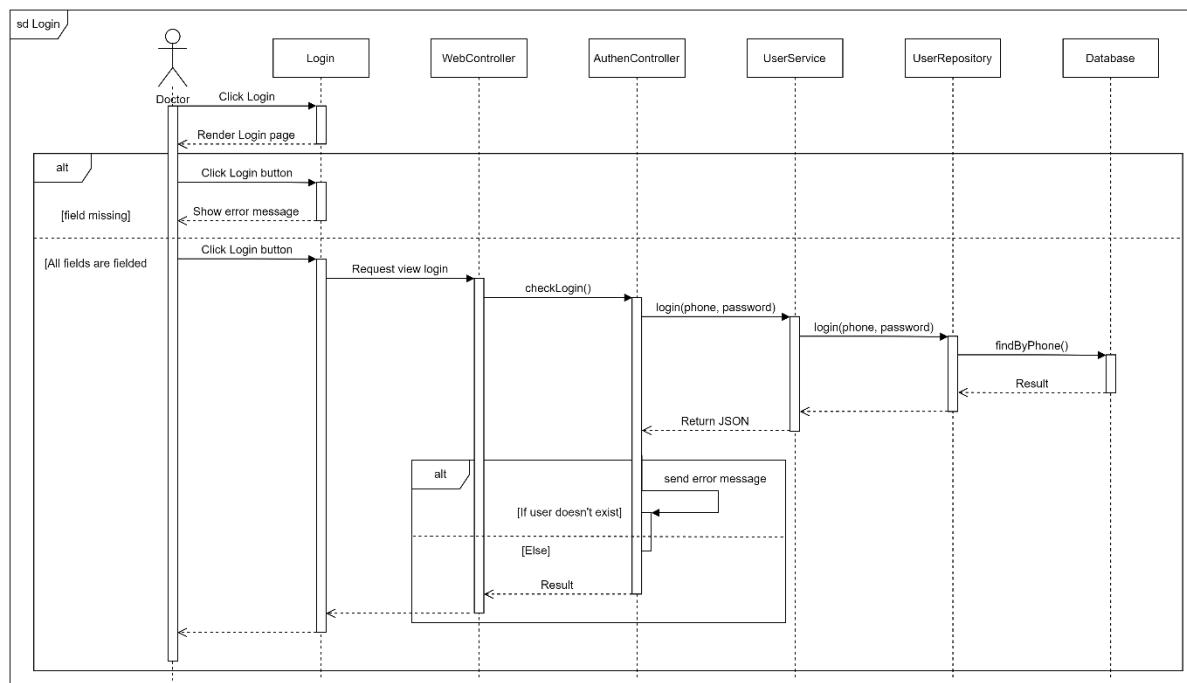
3. Detailed Design

3.1 Login

3.1.1 Class Diagram

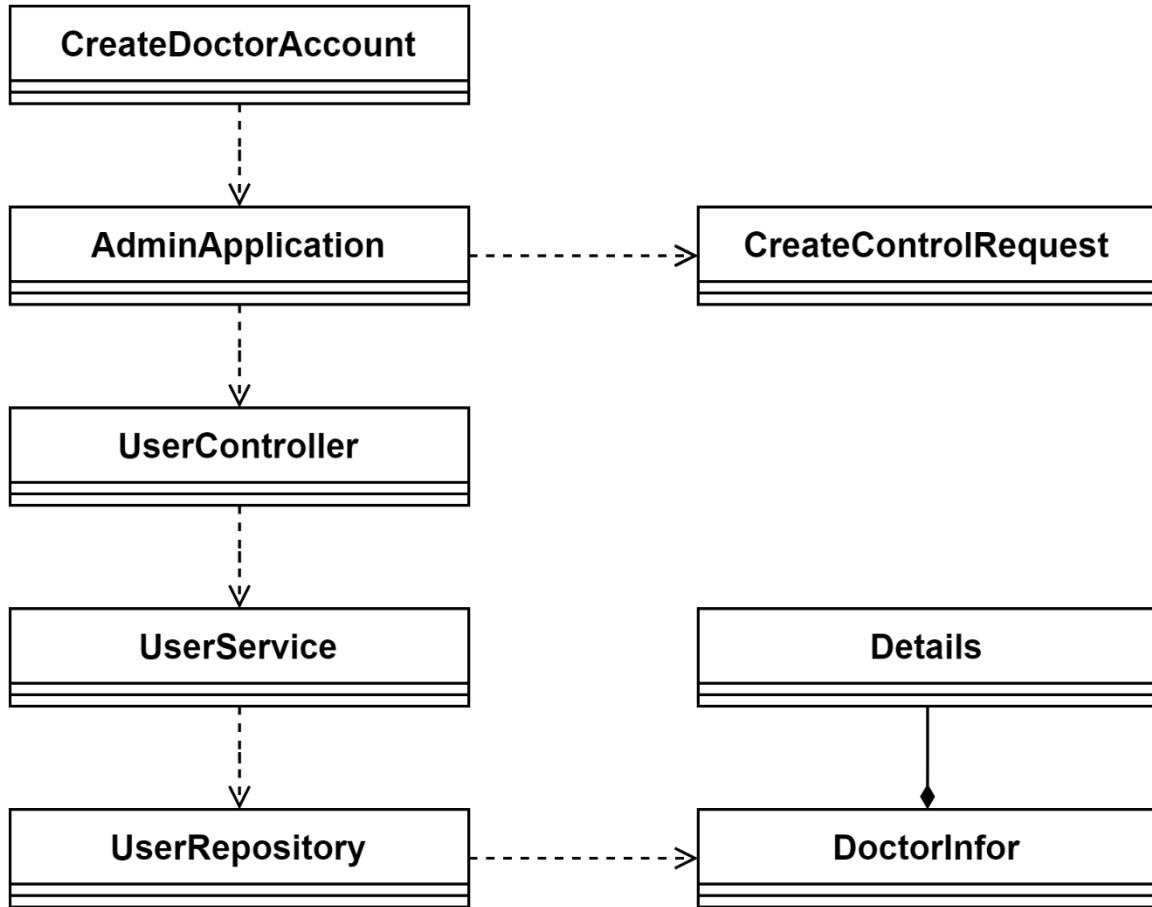


3.1.2 Sequence Diagram Login

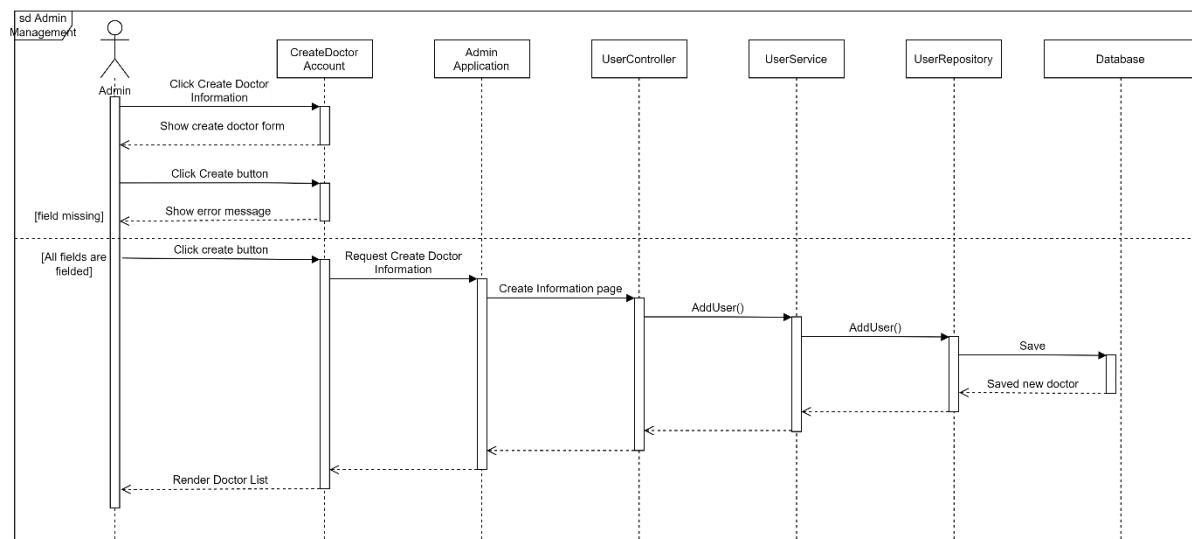


3.2 Admin Management (Create)

3.2.1 Class Diagram CreateDoctorAccount

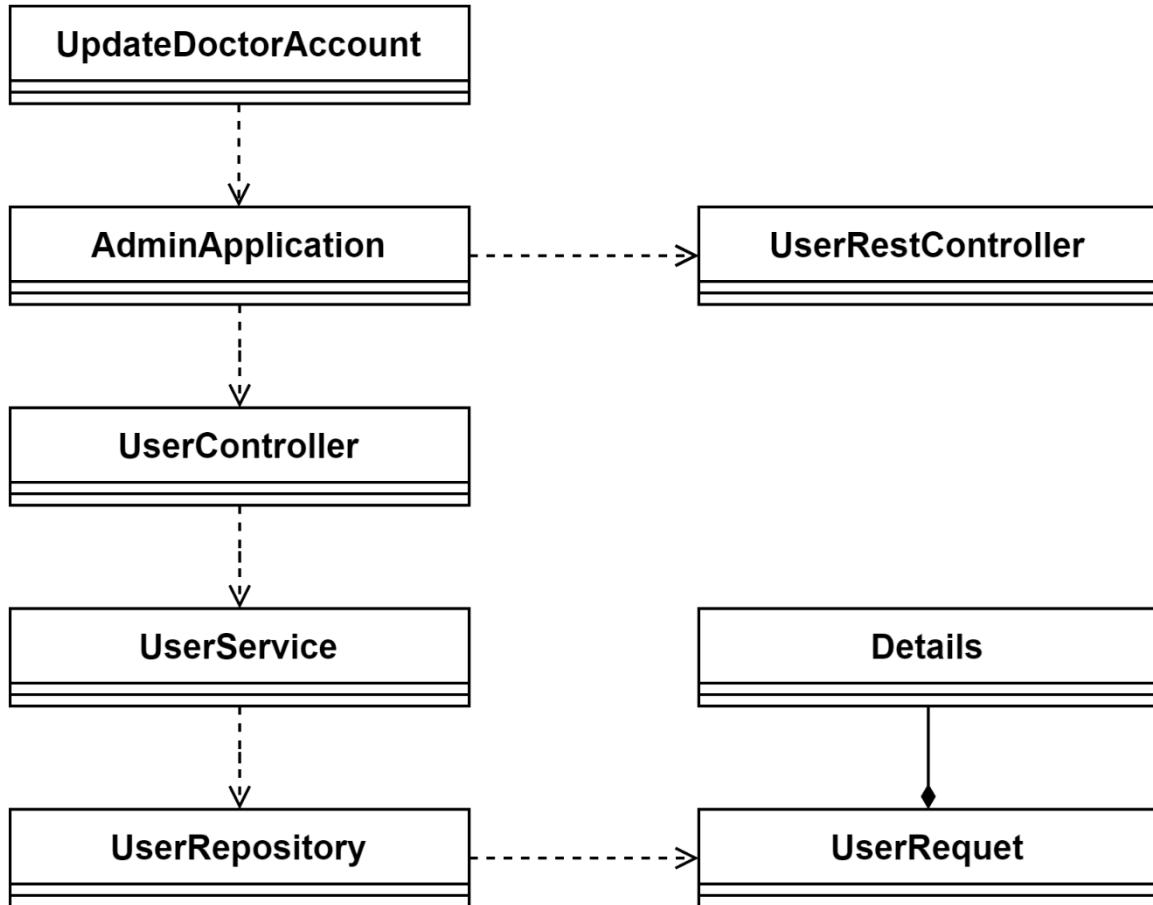


3.2.2 Sequence Diagram CreateDoctorAccount

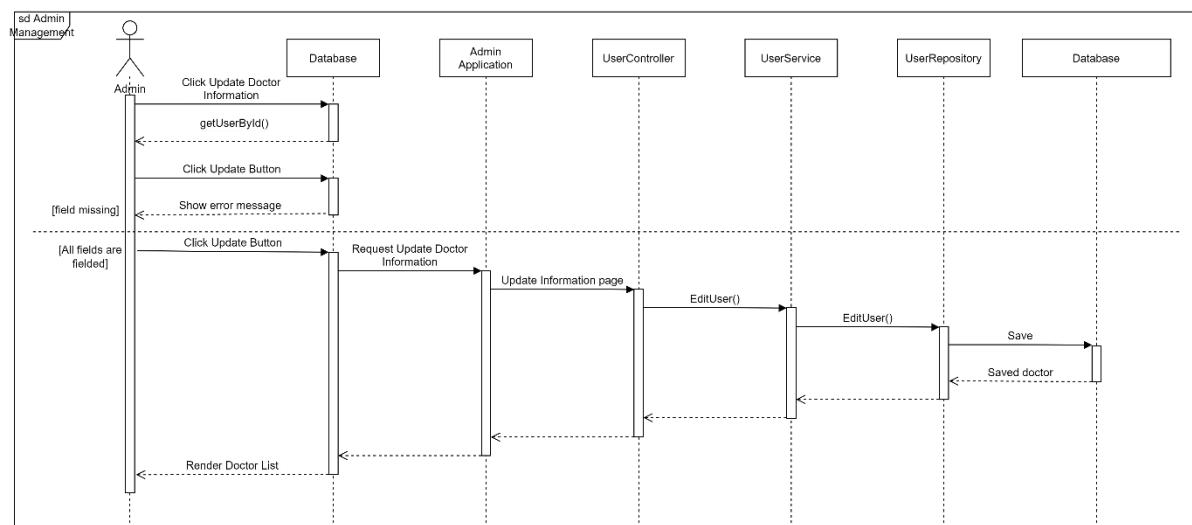


3.3 Admin Management (Update)

3.3.1 Class Diagram UpdateDoctorAccount

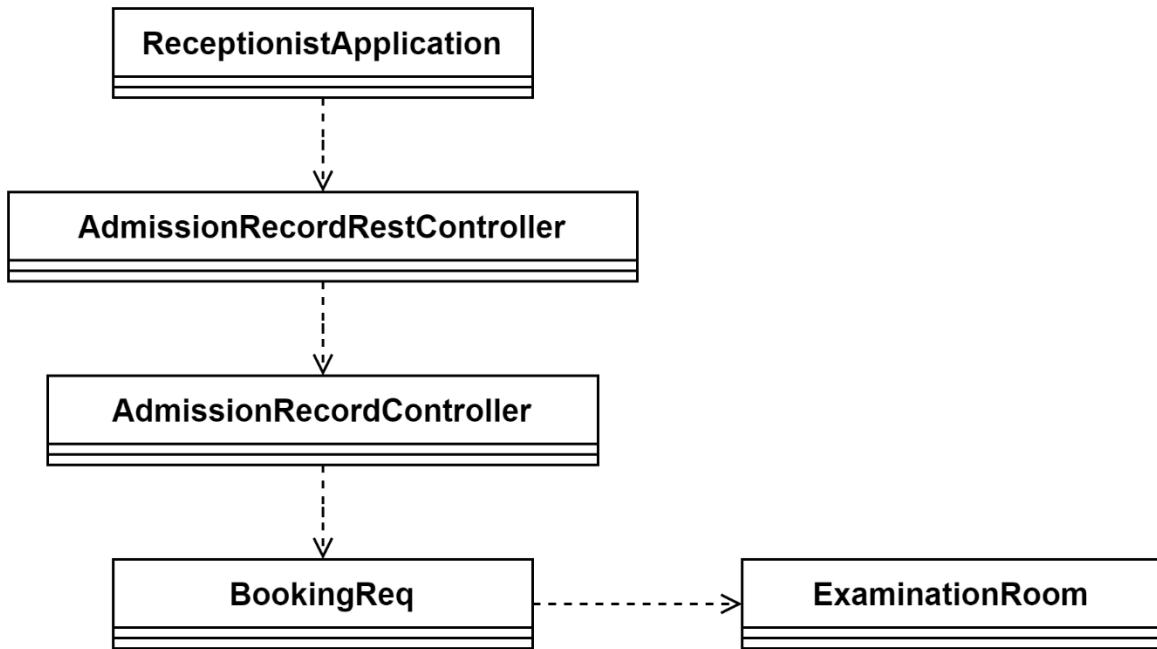


3.3.2 Sequence Diagram UpdateDoctorAccount

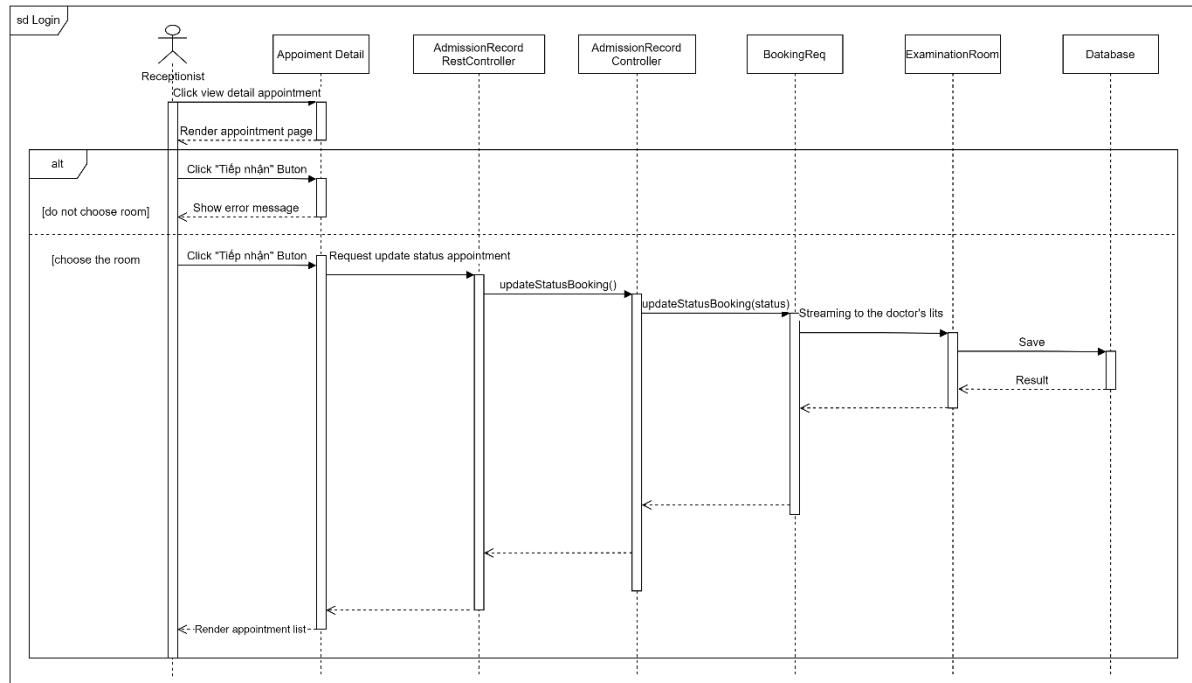


3.3 Admin Management (Delete)

3.3.1 Class Diagram Delete DoctorAccount

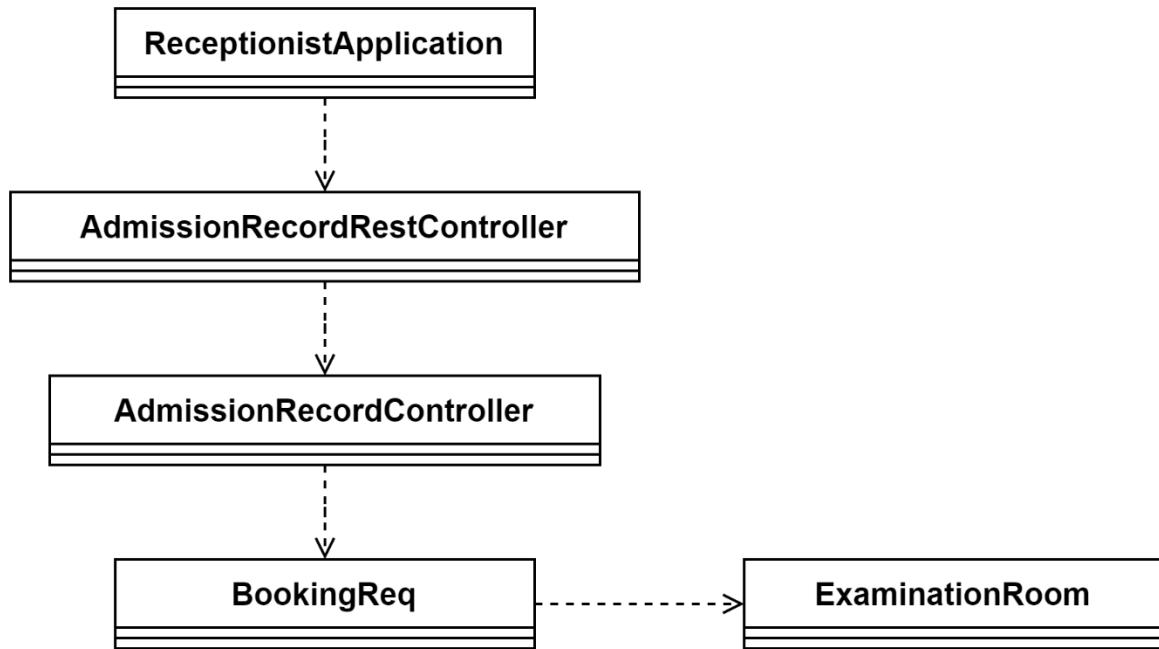


3.3.2 Sequence Diagram Delete DoctorAccount

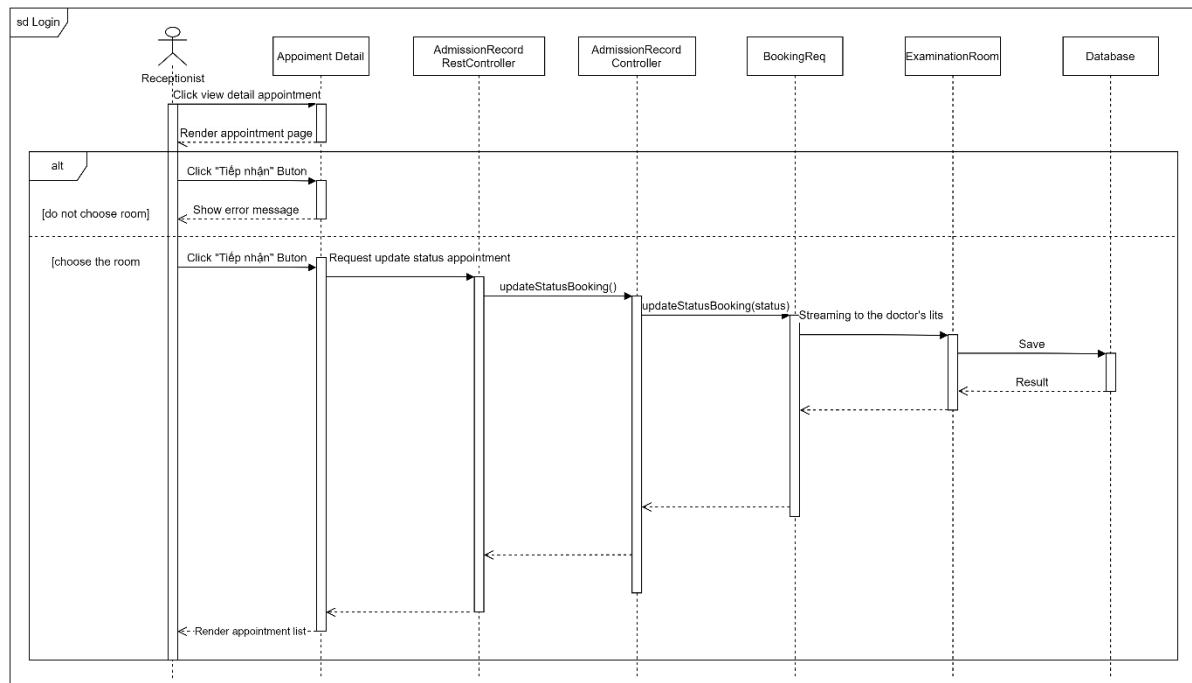


3.4 Threading Patients

3.4.1 Class Diagram Threading Patients

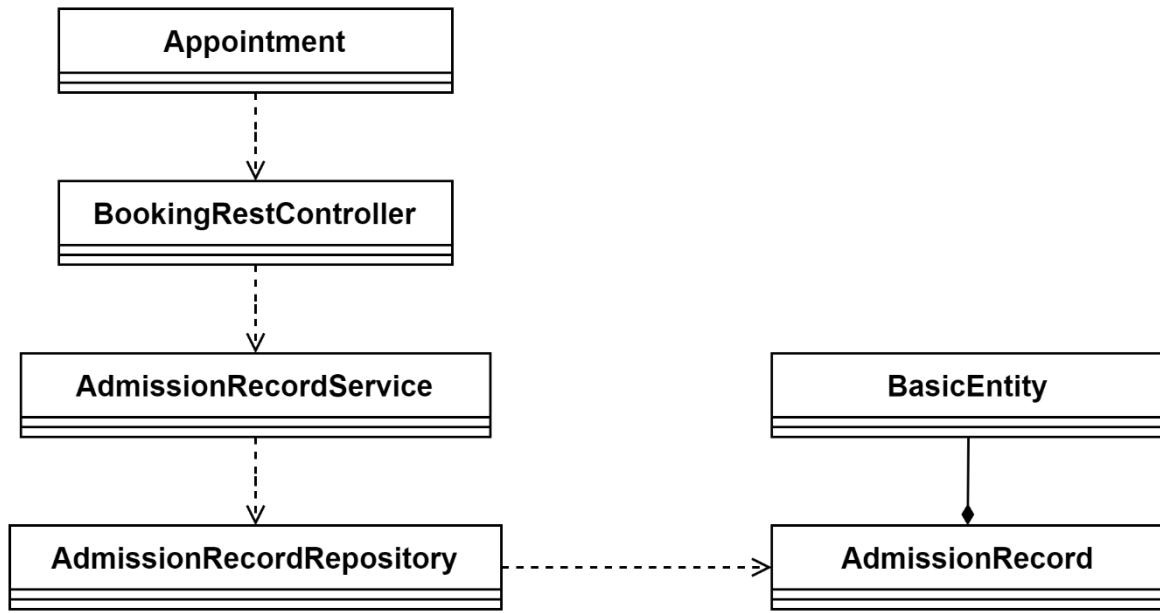


3.4.2 Sequence Diagram Threading Patients

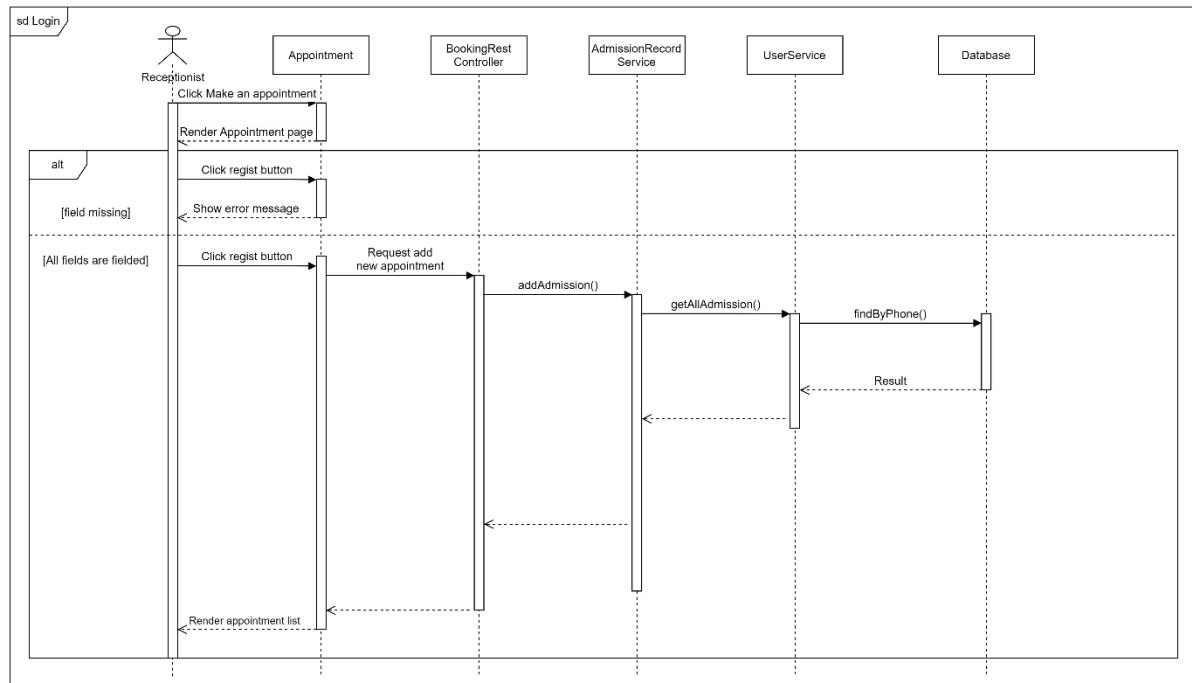


3.5 Make Appointment

3.5.1 Class Diagram MakeAnAppointment

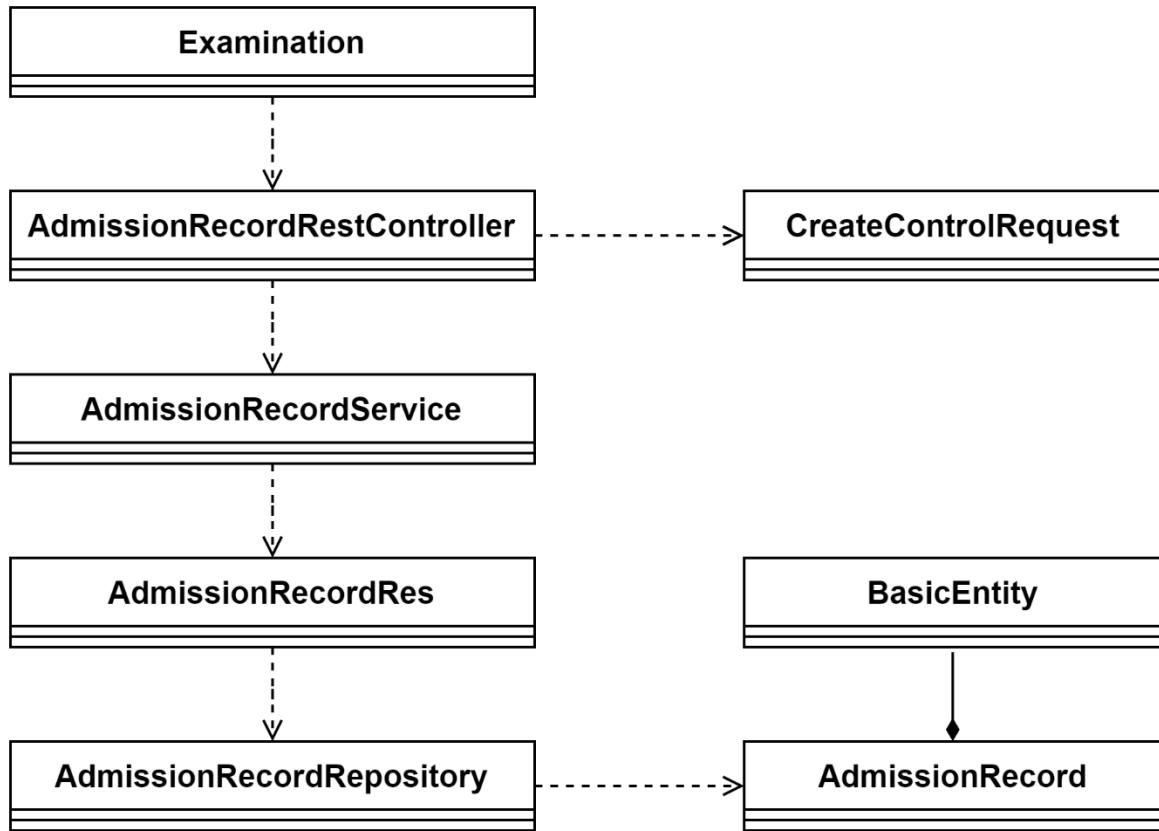


3.5.2 Sequence Diagram MakeAnAppointment

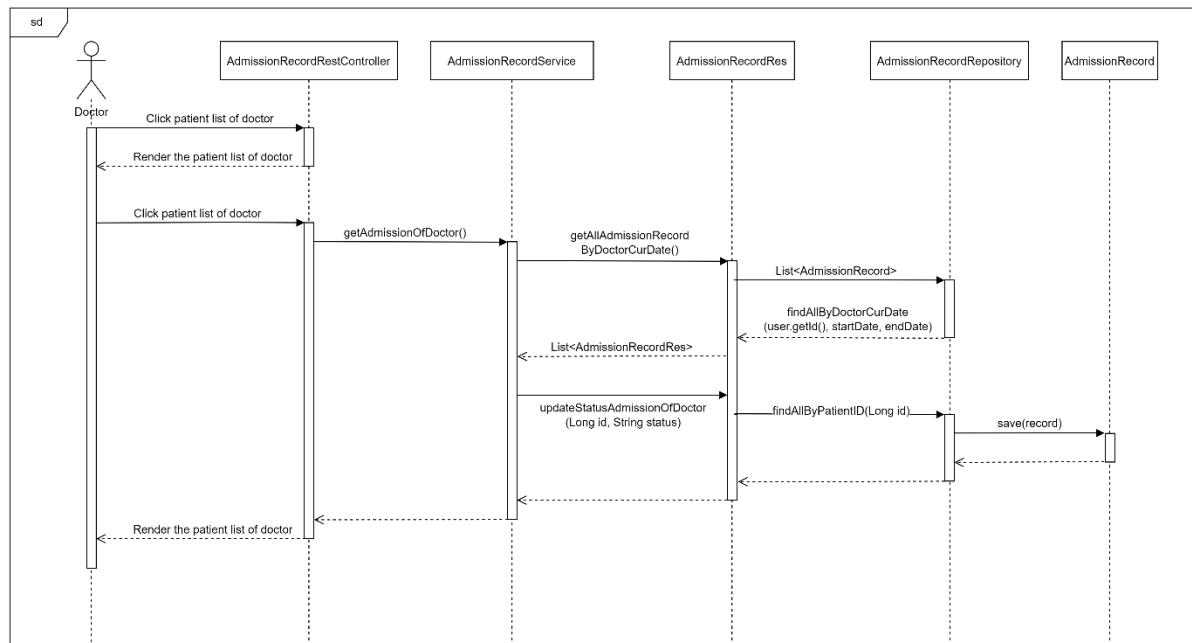


3.6 Doctor Examination

3.6.1 Class Diagram Doctor Examination

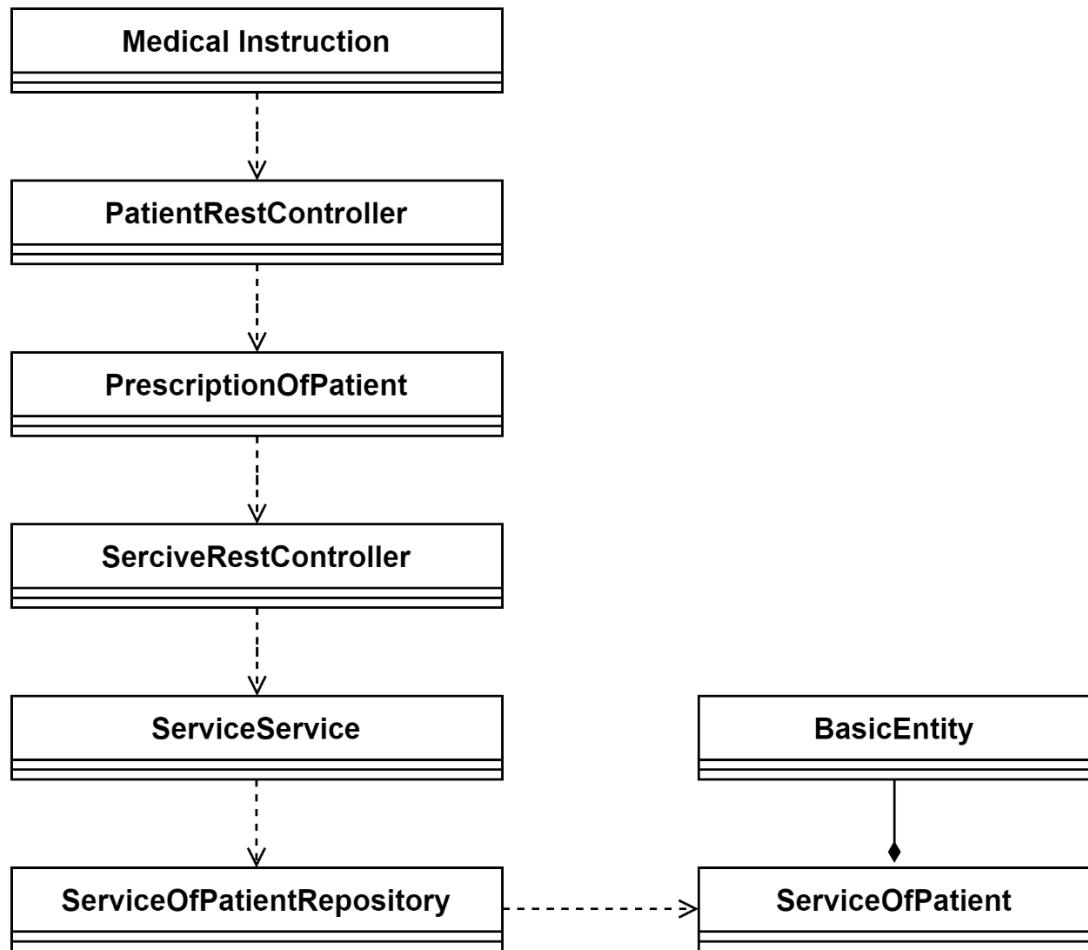


3.6.2 Sequence Diagram Doctor Examination

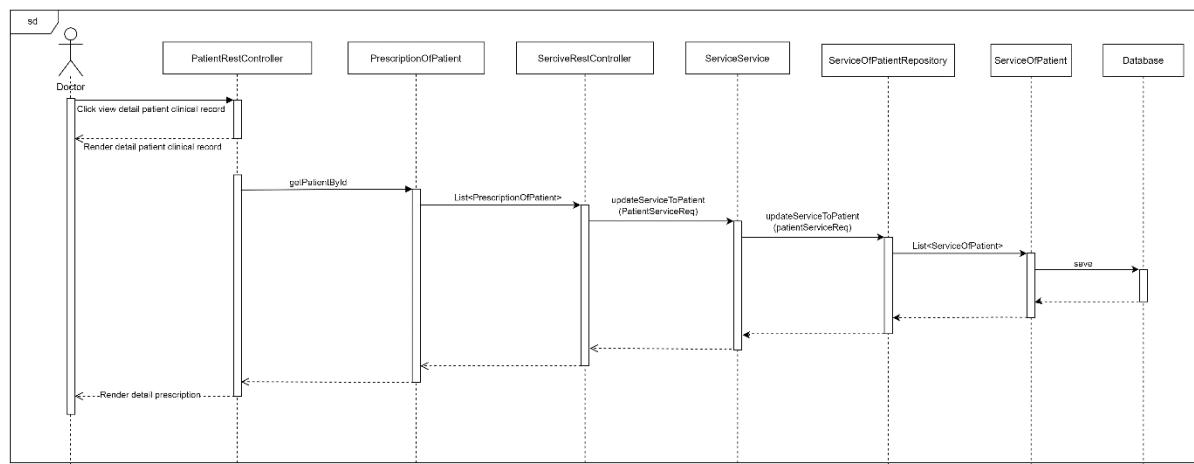


3.7 Medical Instruction

3.7.1 Class Diagram Medical Instruction

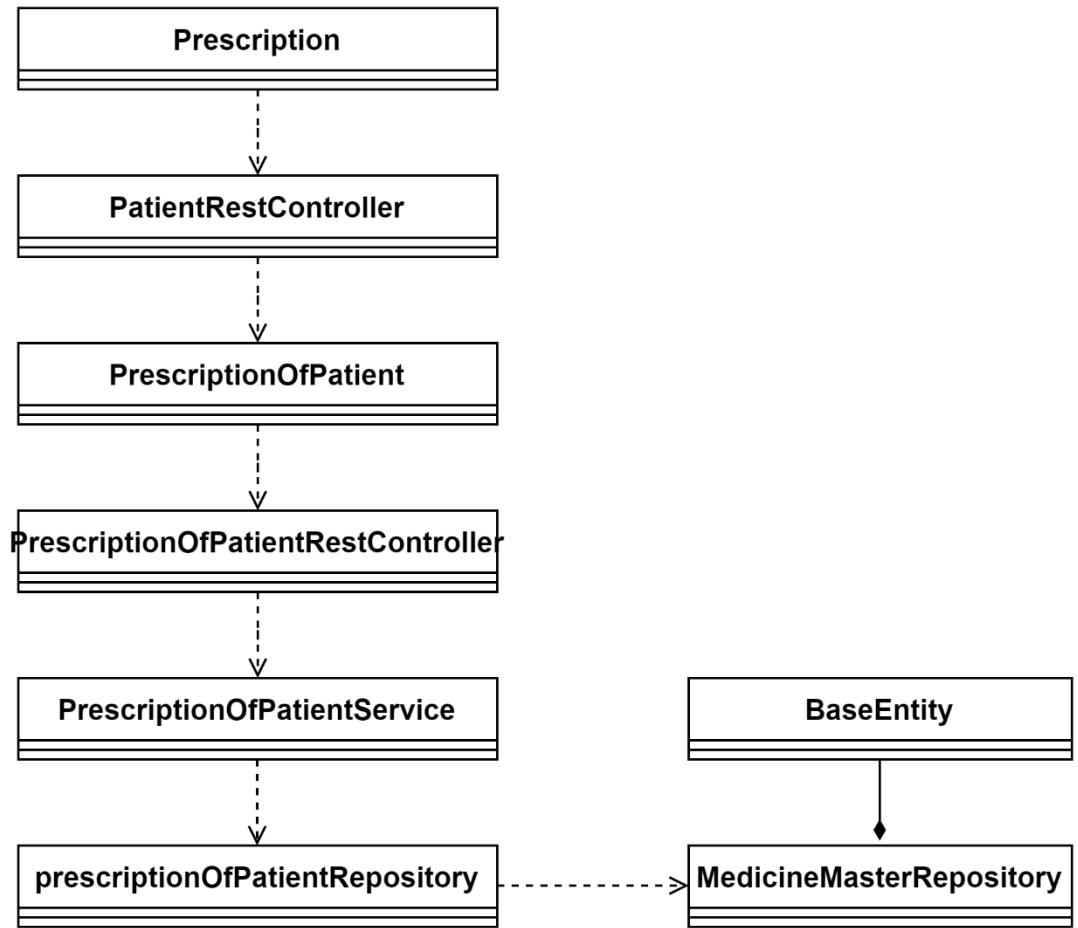


3.7.2 Sequence Diagram Medical Instruction

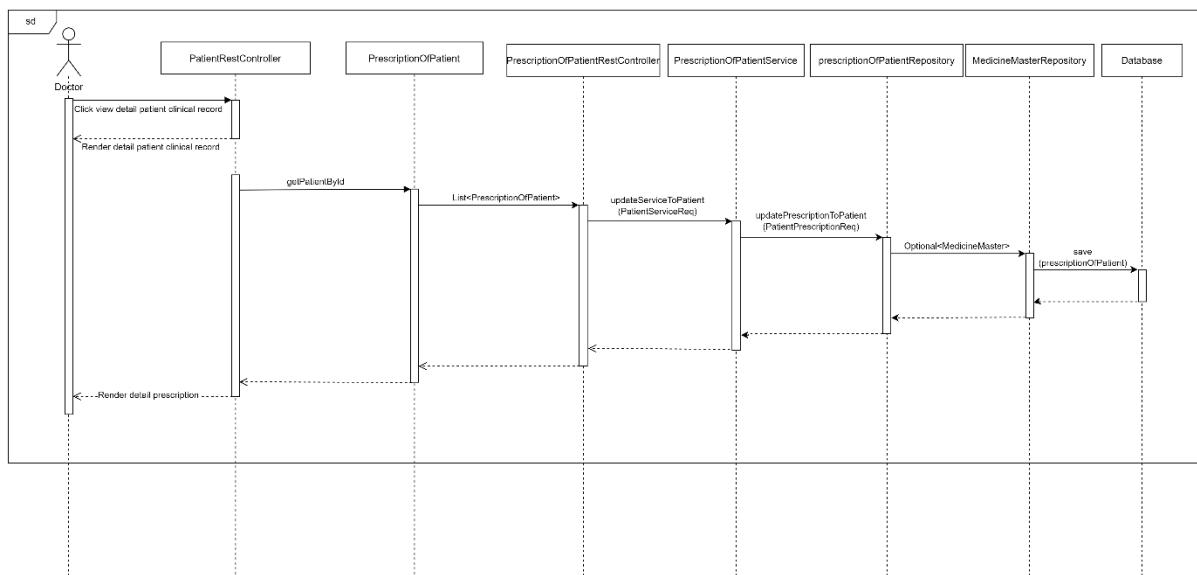


3.8 Print Prescription

3.8.1 Class Diagram Print Prescription

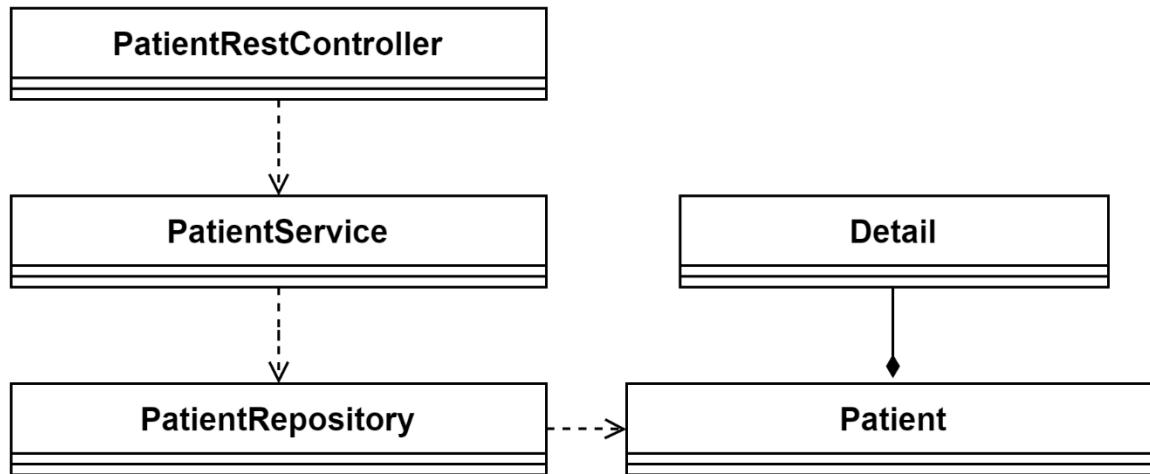


3.8.2 Sequence Diagram Print Prescription

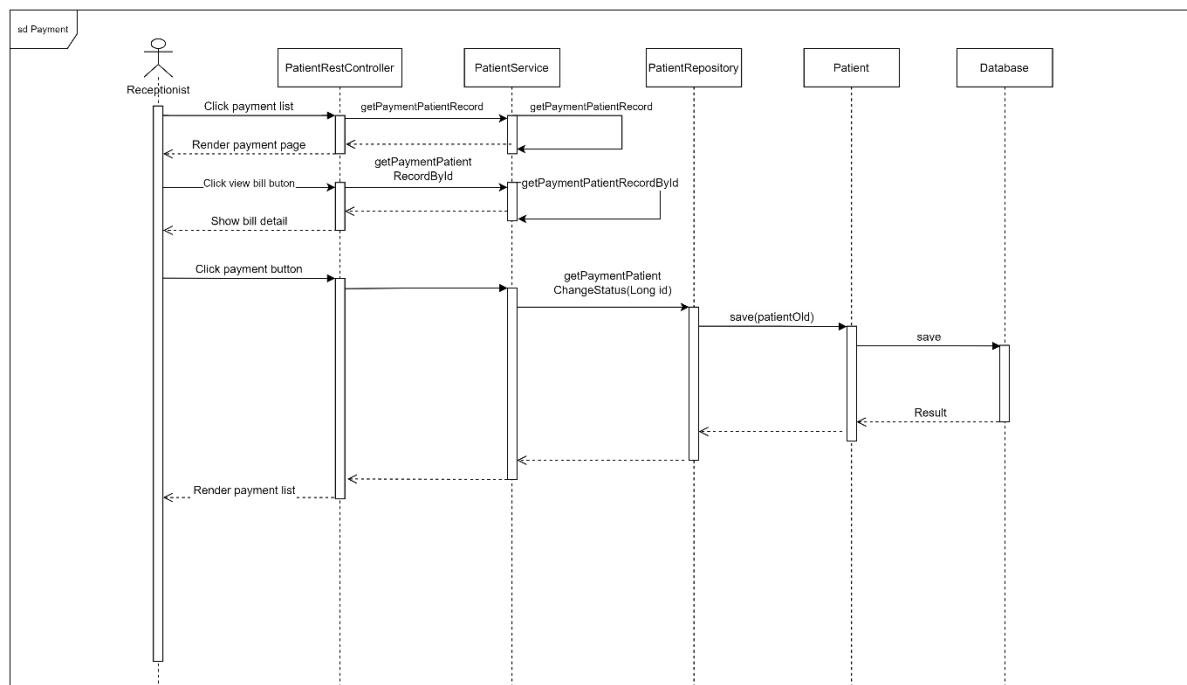


3.9 Payment

3.9.1 Class Diagram Payment



3.9.2 Sequence Diagram Payment

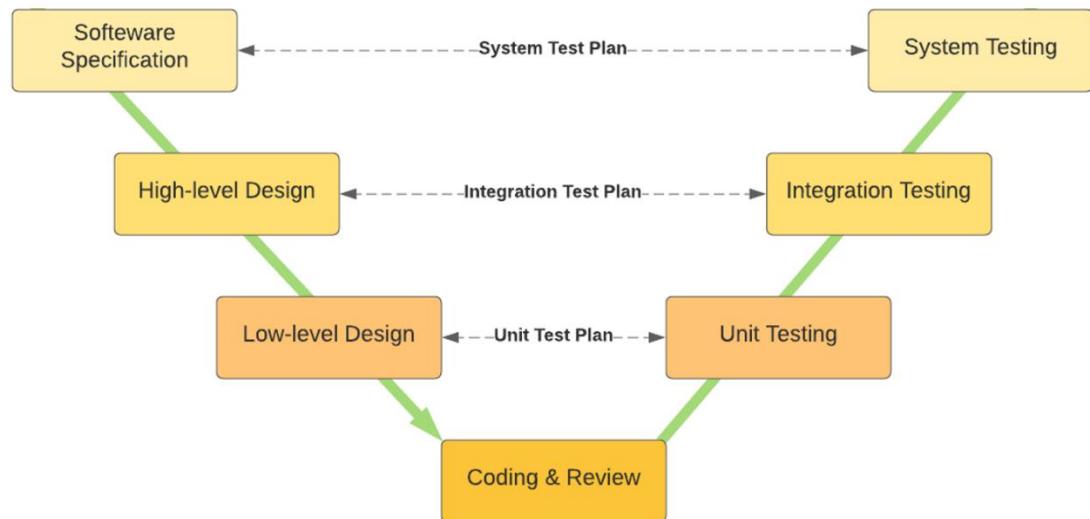


V. Software Testing Documentation

1. Scope of Testing

1.1 Test Model

This project follows the V-model process to implement testing.



No	Verification Phases	Validation Phases
1	Software Requirement Specification: Once you have your clear and detailed product requirements, it's time to complete the system design. The system designer will complete the understanding and detailing of the software and communication setup for the product under development. The system test plan was developed based on the system design. Doing this at an earlier stage leaves more time for the actual test execution afterward.	System Testing: System testing is the first level in which the complete application is tested as a whole. System testing is directly associated with the system design phase. System tests check the entire system functionality and the communication of the system under development with external systems. Most of the software and hardware compatibility issues can be uncovered during this system test execution.

2	<p>Architectural Design: Architectural specifications are understood and designed in this phase. Usually, more than one technical approach is proposed, and based on the technical and financial feasibility the final decision is taken. The system design is broken down further into modules taking up different functionality. This is also referred to as High-Level Design (HLD). The data transfer and communication between the internal modules and with the outside world (other systems) is clearly understood and defined in this stage. With this information, integration tests can be designed and documented during this stage.</p>	<p>Integration Testing: Integration testing allows individuals the opportunity to combine all of the units within a program and test them as a group. This testing level is designed to find interface defects between the modules/functions. This is particularly beneficial because it determines how efficiently the units are running together. Integration testing is associated with the architectural design phase. Integration tests are performed to test the coexistence and communication of the internal modules within the system.</p>
3	<p>Detailed Design: In this phase, the detailed internal Design for all the system modules is specified, referred to as Low-Level Design (LLD). The design must be compatible with the other modules in the system architecture and the other external systems. Unit tests are an essential part of any development process and help eliminate the maximum faults and errors at a very early stage. These unit tests can be Designed at this stage based on the internal module Designs.</p>	<p>Unit Testing: The main aim of this endeavour is to determine whether the application functions as designed. In this phase, a unit can refer to a function, individual program, or even a procedure, and a Whitebox Testing method is usually used to get the job done. Unit tests Designed in the module Design phase are executed on the code during this validation phase. Unit testing is the testing at the code level and helps eliminate bugs at an early stage, though all defects cannot be uncovered by unit testing.</p>

2. Test Strategy

2.1 Testing Types

2.1.1. Function Testing

- The implementation of the functional test will be passed if all functional cases in the Test case document are tested and passed.
- The test case will have to cover all logic branches that function or method could execute with different data input.
- All functional test cases have been executed to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the business rules, and passed.

- The appropriate activities will be performed when valid data is used
- The corresponding error/warning message mechanism is applied for each specific case.

2.1.2. API Testing

- API testing will test all of the individual implemented API of Back-end Service.
- Test case will verify constraints of data which are mentioned in the Business rule.
- All API will be tested with Swagger and Postman tools

2.1.3. UI/UX Testing

- UI/UX tests will be performed fully on all screens.
- UI/UX test targets to cover the verification of the overall look and feel of the system including initial position, font, text size, color, focus, initial button, tab order, label, screen sizes, sentence width, and animation.
- Check all the UI elements for color, size, position, width, length and
- acceptance of characters or numbers
- All GUI test cases have been executed and passed.
- Any defects related to GUI are fixed, except those accepted by the customer.

2.2 Test Levels

Regarding the testing phase, there are four main software testing phases:

2.2.1. Unit Testing

A Unit is a smallest testable portion of a system or application that can be compiled, linked, loaded, and executed. This kind of testing helps to test each module separately. The aim is to test each part of the software by separating it. It checks that components are fulfilling functionalities or not. This kind of testing is performed by developers.

Unit Testing



During this first round of testing, the program is submitted to assessments that focus on specific units or components of the software to determine whether each one is fully functional. The main aim of this endeavor is to determine whether the application functions as Designed. In this phase, a unit can refer to a function, individual program, or even a procedure, and a White-box Testing method is usually used to get the job done. One of the biggest benefits of this testing phase is that it can be run every time a piece of code is changed, allowing issues to be resolved as quickly as possible. It's quite common for software developers to perform unit tests before delivering software to testers for formal testing.

You can reference at <https://viblo.asia/p/unit-test-co-ban-jvElaGGoKkw>

2.2.2. Integration Testing

Integration means combining. For example, in this testing phase, different software modules are combined and tested as a group to make sure that the integrated system is ready for system testing.

Integrating testing checks the data flow from one module to other modules. This kind of testing is performed by testers.



Integration testing allows individuals the opportunity to combine all of the units within a program and test them as a group. This testing level is Designed to find interface defects between the modules/functions. This is particularly beneficial because it determines how efficiently the units are running together. Keep in mind that no matter how efficiently each unit is running, if they aren't properly integrated, it will affect the functionality of the software program. To run these types of tests, individuals can make use of various testing methods, but the specific method that will be used to get the job done will depend greatly on how the units are defined.

You can reference at <https://viblo.asia/p/tim-hieu-ve-integration-testing-QpmleJMo5rd>

2.2.3. System Testing

System testing is performed on a complete, integrated system. It allows checking the system's compliance as per the requirements. It tests the overall interaction of components. It involves load, performance, reliability, and security testing. System testing most often the final test to verify that the system meets the specification. It evaluates both functional and non-functional needs for the testing.



System testing is the first level in which the complete application is tested as a whole. The goal at this level is to evaluate whether the system has complied with all of the outlined requirements and to see that it meets Quality Standards. System testing is undertaken by independent testers who haven't played a role in developing the program. This testing is performed in an environment that closely mirrors production. System Testing is very important because it verifies that the application meets the technical, functional, and business requirements that were set by the customer.

You can reference at <https://viblo.asia/p/system-testing-kiem-thu-he-thong-aWj53pOPK6m>

2.3 Supporting Tools

N/A

3. Test Plan

3.1 Human Resources

Worker/Doer	Role	Specific Responsibilities/Comments
TriBHM	Test Leader/Tester	Manage test resources and assign test tasks Create Test Plan, Test Cases (IT, ST), Test Scripts (IT, ST) Review Test Data Final Inspection Test Cases Create Test Reports
DuyLC	Project Manager/Tester	Approve Test Cases (UT, IT, ST), Test Plan, Test Results, Test Reports
LinhNH, QuanVV	Developer	Create UT Cases, UT Reports

3.2 Test Environment

Purpose	Tool	Provider	Version
Unit Testing	Visual Studio Code IntelliJ IDEA	Microsoft Corporation JetBrains	1.55.2 2021.1
API Testing	Postman	Postman, Inc	8.2.3
User Interface Testing	Chrome	Google	90.0.4430.93
Integration Testing	Chrome	Google	90.0.4430.93
System Testing	Chrome	Google	90.0.4430.93

3.3 Test Milestones

Milestone Task	Effort (MD)	Start Date	End Date
Integration 1:	10	5/3/2022	16/3/2022
Test Plan	2	5/3/2022	16/3/2022
Unit Testing, API Testing	3	6/3/2022	16/3/2022
Integration Testing	5	6/3/2022	16/3/2022
Integration 2:	20	17/3/2022	30/3/2022
Test Plan	2	17/3/2022	30/3/2022
Unit Testing, API Testing	4	18/3/2022	30/3/2022
Integration Testing	14	18/3/2022	30/3/2022
Integration 3:	20	31/3/2022	12/4/2022
Test Plan	2	31/3/2022	12/4/2022
Unit Testing, API Testing	4	1/4/2022	12/4/2022
Integration Testing	14	1/4/2022	12/4/2022
System Testing	8	6/4/2022	14/4/2022

4. Test Cases

4.1 Unit Testing

	input 1: idPrescriptions = 2; ;In this case	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	input 2: idPrescriptions = null; ;In this case	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓
	input 3: idPrescriptions = -10000(id is not exist) ; ;In this case	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓
	input4: idPrescriptions = "!@#\$%^&*(" ; ;In this case	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓
	input 5: idPrescriptions = "abc" ; ;In this case	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓
	input 6: idPrescriptions = 2; ;In this case	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓
	input 7: idPrescriptions = 2; ;In this case	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓
	input8: idPrescriptions = 2; ;In this case	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
	input9: idPrescriptions = 2; ;In this case	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗
	input10: idPrescriptions = null; ;In this case	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗
Confirm	Return	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	2000OK	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓
	FAILL	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Exception		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Request FAIL	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	Internal server error	✓	✓	✓	✓	✗	✗	✓	✓	✓	✓
		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Log message		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Success	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓
		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Result	Type(N : Normal, A : Abnormal, B : Boundary)	N	A	A	A	A	A	A	A	A	A
	Passed/Failed	P	P	P	P	P	P	P	P	P	P
	Executed Date	0	0	0	0	0	0	0	0	0	0

Figure 1: Unit Test Case Example

Function	Speed	Static	Status
editUserLogin()	70ms	user-management.js	200 OK
deleteUserAction()	181ms	user-management.js	200 OK
addEditUserAction()	115ms	user-management.js	200 OK
getUserByPage(dataSearch)	76ms	user-management.js	200 OK
editUser(id)	153ms	user-management.js	200 OK
getPatientByPage(dataSearch)	69ms	profile-patient-management.js	200 OK
deleteCatalogAction()	196ms	service-management.js	200 OK
addEditServiceAction()	280ms	service-management.js	200 OK
deleteCatalogAction()	582ms	icd10-management.js	200 OK
addEditICDAAction()	1130ms	icd10-management.js	200 OK
editICD(id)	118ms	icd10-management.js	200 OK
getICDByPage(dataSearch)	131ms	icd10-management.js	200 OK
deleteCatalogAction()	508ms	medicine-management.js	200 OK
addEditMedicineAction()	101ms	medicine-management.js	200 OK
editMedicine(id)	217ms	medicine-management.js	200 OK

Figure 2: API Test Speed Example

4.2 Integration Testing & System Testing

The term “integration” means a process of combining into an integral whole. Similarly, integration testing of a build indicates separate testing units of a system as one.

A system test inspects every software unit to secure their proficiency as a whole or assembled build. In software engineering, a system test takes place after unit and integration testing.

Integration Test and System Test are done by testers to ensure that combined units work correctly and that the system functions as intended.

GUI testing is also done during this process to ensure that elements and functions load correctly, the text is readable, and the website interface looks good in various browser sizes.

This is the sample tests of Integration Test and System Test in CIS system:

Detailed test cases of these types of testing are described in 2 files: CIS_IntegrationTest.xlsx (for Integration Testing) and CIS_SystemTest.xlsx (for System Testing).

Test Case ID	Test Case Description	Test Case Procedure	Expected Results	Pre-conditions	Round 1	Test date	Tester
[Admin-Categories Management-1]	Test GUI "Danh mục dịch vụ kỹ thuật" form	1. Login the system with Admin role. 2. Click "Danh mục dịch vụ kỹ thuật" tab in the left menu.	The "Danh mục dịch vụ kỹ thuật" form displayed with the following information: - STT - Chuyên Khoa - Mã tương đương - Tên theo Danh mục tại thông tư 43,50,21 - Phân Tuyển - Phân loại PTIT - Chuyên khoa theo thông tư 43	Connect to database success	Passed	6/3/2022	TriBHM
[Admin-Categories Management-2]	Test function viewing "Danh mục dịch vụ kỹ thuật" page with many categories	1. Login the system with Admin role. 2. Click "Danh mục" tab in the left menu. 3. Click "Danh mục dịch vụ kỹ thuật"	Display categories list with informations paging.	Connect to database success	Passed	6/3/2022	TriBHM
[Admin-Categories Management-3]	Test GUI viewing pagination "Danh mục dịch vụ kỹ thuật" page with too many categories	1. Login the system with Admin role. 2. Click "Danh mục dịch vụ kỹ thuật" tab in the left menu.	Display 10 categories in list each page	Connect to database success	Passed	6/3/2022	TriBHM
[Admin-Categories Management-4]	Test GUI "Danh mục bệnh ICD10" form	1. Login the system with Admin role. 2. Click "Danh mục bệnh ICD10" tab in the left menu.	The "Danh mục bệnh ICD10" form displayed with the folowing information: - STT - Sít Chuong -Mã Chuong -Chapter name -Tên Chuong -Mã nhóm chính -Tên nhóm chính -Tên loại -Mã bệnh -Disease Name -Tên bệnh	Connect to database success	Passed	6/3/2022	TriBHM

Patient-Appointment Receptionist - Patient Management Admin- Categories management Admin - Doctor Management Authentication Admin - Patient Management

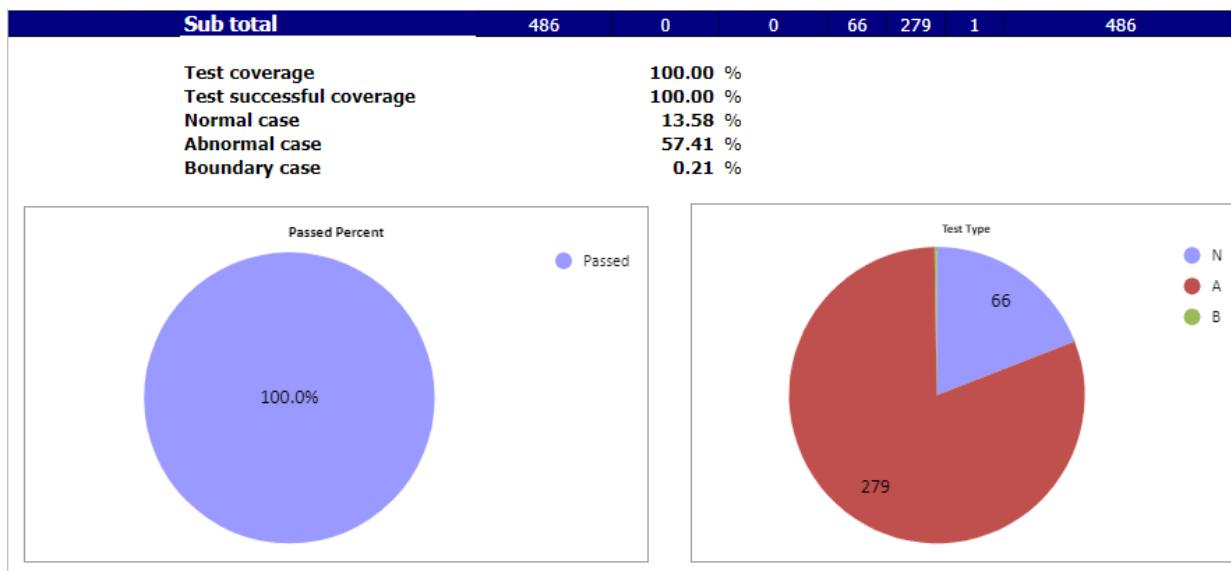
Figure 1: Integration Test Case Example

ID	Test Case Description	Test Case Procedure	Expected Results	Inter-test case Dependence	Result	Test date	Tester	Note
1	Patient make new appointment	1.Click buton "Đăng ký khám" from homepage 2. Fill all the requirement to the appointment form 3. click buton "Đăng ký"	Booking successfully.		Pass	10/04/2022	TriBHM	
2	Reception accepts appointment if patient already exists in system	1.Select "Danh sách lịch hẹn" tab in the left menu 2. search by phone number 3. Click buton "xem" 4. click buton " Xác nhận lịch hẹn"	Accept successfully Status is confirm	Login with receptionist role	Pass	10/04/2022	TriBHM	
3	Receptionist threading patient to the doctor room	1.Select "Danh sách lịch hẹn" tab in the left menu 2. Click buton "xem" 3. Select room 4. Click "xác nhận"	Recive successfully Status is receive Data is update for "Danh sách bệnh nhân"	Login with receptionist role	Pass	10/04/2022	TriBHM	
4	Show patient list of the receiving doctor	1. Select "Danh sách bệnh nhân" tab in the left menu	Display 4 patient list	Login with doctor role	Pass	10/04/2022	TriBHM	
5	Receiving patients from the waiting room for medical examination	1. Select patient from " Danh sách chờ" 2. Click "Khám bệnh"	Move patient from "Danh sách bệnh nhân chờ" to "Danh sách bệnh nhân đang khám"	Login with doctor role	Pass	10/04/2022	TriBHM	
6	View the patient clinical record	1. Select "Danh sách bệnh nhân" tab in the left menu 2. Select patient from " Danh sách chờ" 3. Click "Xem"	Display patient clinical record with vital signs detail and history examination	Login with doctor role	Pass	10/04/2022	TriBHM	
7	Measure the patient's vitals	1. Fill the patient's vital 2. Click "lưu thông tin"	Save the patient's vitals	Login with doctor role	Pass	10/04/2022	TriBHM	
8	Perform medical	1. fill in the diagnosis information		Login with doctor role	Pass			

Figure 2: System Test Case Example

5. Test Reports

5.1 Unit Test Report



5.2 Integration Test Report

No	Module code	Passed	Failed	Pending	N/A	Number of test cases
1	Authentication	21	0	0		21
2	Patient-Appointment	5	0	0		5
3	Reception-Appointment	17	0	0		17
4	Receptionist - Patient Management	19	0	0		19
5	Admin - Categories Management	54	0	0		54
6	Admin - Doctor Management	19	0	0		19
7	Admin - Receptionist Management	19	0	0		19
8	Admin - Patient Management	19	0	0		19
9	Manage Clinical Record	6	0	0		6
10	Manage Clinical Record(Service)	17	0	0		17
11	Manage Clinical Record(Medicine)	17	0	0		17
12	Admin-Manage Doctor Schedule	20	0	0		20
13	Doctor- Pre-Condition	8	0	0		8
14	Doctor- Treatment Regimen Management	50	0	0		50
15	Doctor - Profile	9	0	0		9
Sub total		300	0	0	0	300

Test coverage **100.00 %**
Test successful coverage **100.00 %**

5.3 System Test Report

No	Module code	Passed	Failed	Pending	N/A	Number of test cases
1	Scenario 1	14	0	0	0	14
2	Scenario 2	2	0	0	0	2
3	Scenario 3	19	0	0	0	19
4	Scenario 4	19	0	0	0	19
5	Scenario 5	21	0	0	0	21
6	Scenario 6	19	0	0	0	19
7	Scenario 7	20	0	0	0	20
Sub total		114	0	0	0	114
Test coverage			100.00	%		
Test successful coverage			100.00	%		

5.4 Total LOC

Statistic		Statistic			
		Refresh Refresh on selection Settings			
Overview	css	html	java	js	properties
Extension ▲		Lines CODE			
 css (CSS files)		47859			
 eot (EOT files)		6043			
 html (HTML files)		3373			
 java (Java classes)		6148			
 js (JS files)		37364			
 map (MAP files)		10			
 properties (Java properties files)		21			
 svg (SVG files)		24			
 ttf (TTF files)		6040			
 txt (Text files)		177			
 woff (WOFF files)		1536			
 woff2 (WOFF2 files)		1268			
 xml (XML configuration file)		168			

Java: 8081 LOC

JS: 4305 LOC

Html: 4035 LOC

Total: 16421 LOC

Average 100 LOC has 3 Unit Test Case

VI. Release Package & User Guides

1. Deliverable Package

1.1 Source codes & documents

No.	Items	Sub-Items	Type	Version
Code Package				
1	Front-end	client	New	1.0
2	Back-end	api	New	1.0
Documents				
1	Final report	Clinic Information System_FinalReport.pdf	New	1.0
2	Report 1	Report1_Project Introduction.docx	New	1.0
3	Report 2	Report2_Project Management Plan.docx	New	1.0
4	Report 3	Report3_Software-Requirement-Specification.docx	New	1.0
5	Report 4	Report4_Software Design Document.docx	New	1.0
6	Report 5	Report5_Test Documentation.docx	New	1.0
7	Report 6	Report6_Software-User-Guides.docx	New	1.0
8	CIS_SystemTest	CIS_SystemTest.xlsx	New	1.0

9	CIS_IntegrationTest	CIS_IntegrationTest.xlsx	New	1.0
---	---------------------	--------------------------	-----	-----

1.2 Known Issues, Limitations & Restrictions

N/A

2. Installation Guides

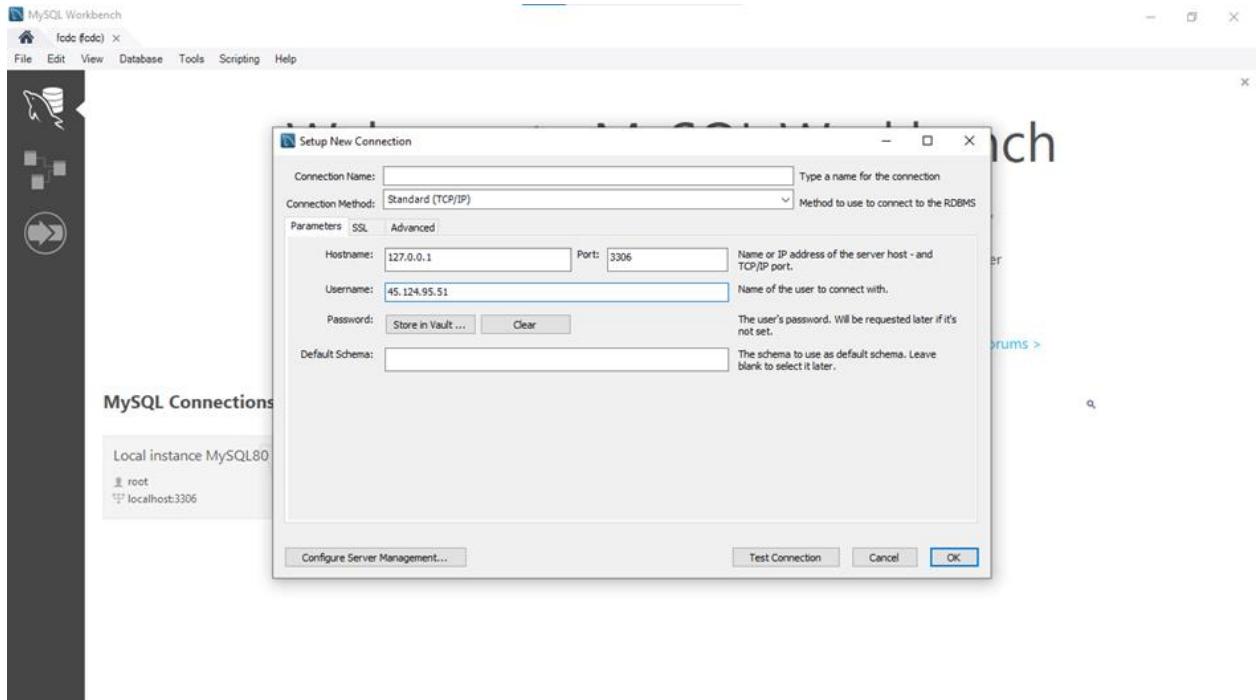
2.1 System Requirements

- Windows 10 Professional 64 bit, Intel® Core™ i5-7200U, RAM 8.00GB
- MySql
- Postman v8.2.3
- Visual Studio Code v1.55.2
- IntelliJ IDEA v2021.1

2.2 Installation Instruction

- Download and install MySql by following the instruction at:
<https://dev.mysql.com/downloads/>

 1. Open MySQL Workbench 8.0 CE.
 2. In New Connection



3. Copy and Paste the username: 103.107.183.232
 4. Click Connect.
- Download and install Postman by following the instruction at:
<https://www.postman.com/downloads/>
 1. Open Postman.

2. In the Home menu. Choose Collections, and click to Import.
 3. Choose the file collection end then click “Import”.
- Download and install Visual Studio Code at: <https://code.visualstudio.com/download>
1. Open lcm-system project in Visual Studio Code
 2. Open terminal in the project folder and run:
 - yarn install
 - yarn start
- Download and install IntelliJ IDEA at: <https://www.jetbrains.com/idea/download/other.html>
1. Open lcm-system project in IntelliJ IDEA
 2. Open terminal in the project folder and run:
 - yarn install
 - yarn start

3. User Manual

3.1 System requirements

Internet access required.

Any laptop, pc, table or mobile that is able to use Google Chrome, Mozilla Firefox or other browsers.

3.2 Application Usage

3.2.1 Guest

- View Homepage:
 - Access homepage
 - User can view all information about the website, user can log in

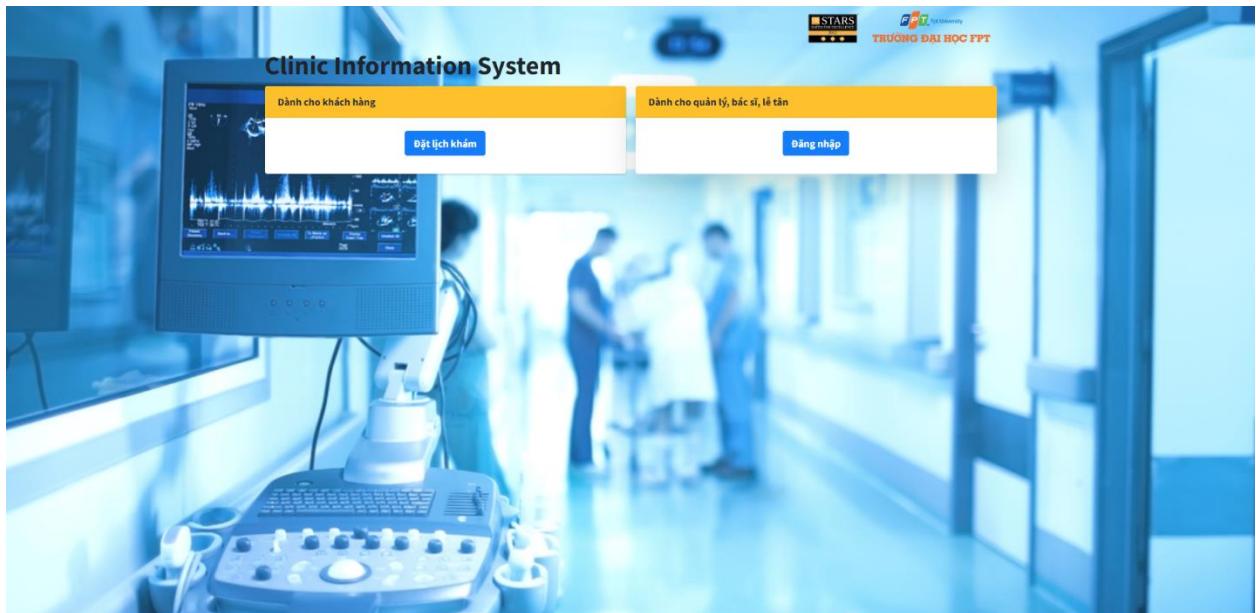


Figure 1: Landing Page Screen

- Login: If the guest is Doctor, Admin, Receptionist
 - On the homepage.
 - Click “Đăng nhập” on the top right of the website as “Dành cho quản lý, bác sĩ, lễ tân”.
 - Input phone number and password and click “Đăng nhập”.



Figure 2: Sign-in screen

3.2.2 Doctor

- View patients list:

- Doctor login
- Click to “Danh sách bệnh nhân” in the left sidebar of website
- Doctor view patients list with their informations.

The screenshot displays the CIS application interface with the following panels:

- Danh sách bệnh nhân đang khám:** Shows two patients: Vũ Văn Quân (Degree of priority: No) and Nguyễn Hữu Huy (Degree of priority: No). Each patient has three buttons: Xem (View), Chờ khám (Wait for examination), and Đã khám (Examined).
- Danh sách bệnh nhân chờ:** Shows one patient: Nguyễn Văn Dũng (Degree of priority: No). It has two buttons: Xem (View) and Khám (Examined).
- Đang chờ kết quả xét nghiệm:** Shows two patients: Nguyễn Hoàng Xuân (Degree of priority: No) and Vũ Công Quyền (Degree of priority: No). Each patient has two buttons: Xem (View) and Đã khám (Examined).
- Bệnh nhân đã khám trong ngày:** Shows two patients: Dương Hào Nam (Degree of priority: No) and Họ và Tên (Degree of priority: No). Each patient has one button: Xem (View).

Figure 3: View patients list

- Update Information:
 - Doctor login
 - Click to name of doctor in in the left sidebar of website
 - Input new information and confirm
 - Click “Lưu” button

Danh sách bệnh nhân

Hồ sơ bệnh nhân

Lịch trực

Điện thoại (*)
0868130599

Mật khẩu (*)

Nhập lại mật khẩu (*)

Họ và tên (*)
Linh Nguyen Hoang

Giới tính (*)
Nam

Ngày sinh (*)
10/05/1999

Email address
doctor@gmail.com

Address
nghe an

Hủy

Lưu

Figure 4: Update Information

- View Treatment regimen

- In the left sidebar of website, doctor click “Danh sách phác đồ”
- The doctor view list of Treatment regimen.
- Doctor can choose a Treatment regimen to view details and edit.

Tạo mới | Tìm kiếm

Phân loại:

Thuốc

Dịch vụ kỹ thuật

Tên chuyên khoa: Riêng Tư

Tên phác đồ:

#	Chuyên khoa	Tên phác đồ	Trạng thái	Người tạo
1	Tai mũi họng	Điều trị viêm tai	Riêng Tư	Linh Nguyen Hoang
2	Mắt	Điều trị mắt	Riêng Tư	Linh Nguyen Hoang

Showing 1 to 2 of 2 entries

Figure 5: View Treatment regimen

- Create Treatment regimen

- In the left sidebar of website, doctor click “Danh sách phác đồ”

- The doctor view list of Treatment regimen.
- Click “Tạo mới” button in the top of website to create new Treatment regimen.
- Input information of new Treatment regimen
- Click “Thêm y lệnh” to confirm create new Treatment regimen.

The screenshot shows a software interface titled 'CIS' with a user profile 'Linh Nguyen Hoang'. The main area is a form for creating a new treatment regimen. At the top right are buttons for 'Save' and 'Tạo lại' (Create again). Below these are input fields for 'Tên phác đồ:' (Treatment Plan) containing 'Khoang mũi', 'Tên chuyên khoa:' (Specialty) containing 'Tai mũi họng', and a dropdown for 'Riêng Tư' (Confidential). The 'Tên dịch vụ:' (Service Name) field contains 'Phẫu thuật nội soi cắt polyp mũi' and a 'Thêm Y lệnh' (Add Prescription) button is visible. A table below lists the prescription details:

#	Chuyên khoa	Y lệnh
	Tai mũi họng	Phẫu thuật nội soi cắt polyp mũi

Figure 6: Create Treatment regimen

- View patient medical records
 - In the left sidebar of website, doctor click “Hồ sơ bệnh nhân”
 - The doctor view list of patients.
 - Doctor can choose a patient to view their medical record details.

The screenshot shows a medical record for patient number 112, Nguyen Huy, born on 01/01/1999, with phone number 0355818489 and gender male. The medical history section indicates a visit on 26/04/2022 for ear pain.

Figure 7: View patient medical records

- View Schedule
 - In the left sidebar of website, doctor click “Lịch trực”
 - The Doctor view their schedule

The screenshot displays a weekly schedule for a doctor named Linh Nguyen Hoang. The schedule is from April 25 to April 29, 2022. It includes two shifts: morning (7:30 - 11:00) and afternoon (14:00 - 17:30). The afternoon shift is shared with another doctor, Hoang Viet.

Năm	Thứ 2	Thứ 3	Thứ 4	Thứ 5	Thứ 6
2022	25/4	26/4	27/4	28/4	29/4
Sáng 7h30 - 11h	Nội Khoa: Linh Nguyen Hoang	Nội Khoa: Linh Nguyen Hoang			
Chiều 14h - 17h30	Phụ Sản: Hoàng Việt	Khoa Tai - Mũi - Họng: Ngô Trúc Vy			
		Phụ Sản: Vũ Văn Quân			

Figure 8: View Schedule

- Logout:
 - In home page
 - Click to “Đăng xuất” on the top right of the website.

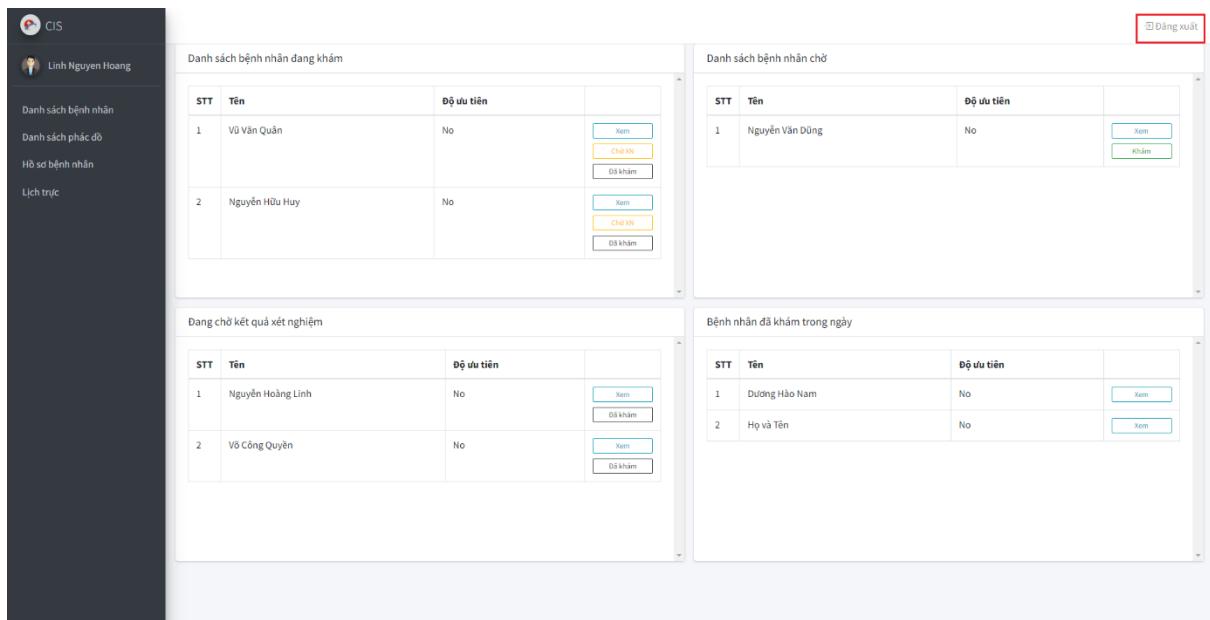


Figure 9: Logout screen

3.2.3 Patient

- Make an appointment:
 - On the homepage.
 - Click “Đặt lịch khám” as “Dành cho khách hàng” on the top left of the website.
 - Input all information and click “Đăng ký”.

Figure 10: Make an appointment

3.2.4 Admin

- View Categories list

- Admin login.
- Click to “Danh mục” and choose one of categories.
- Admin view list of that Category

#	Chuyên khoa	Tên dịch vụ	Mã tương đương
1	Hồi sức cấp cứu và Chống độc	Thông khí nhân tạo không xâm nhập phương thức CPAP [giờ theo thực tế]	01.0130.0209
2	Hồi sức cấp cứu và Chống độc	Thông khí nhân tạo không xâm nhập phương thức BIPAP [giờ theo thực tế]	01.0131.0209
3	Hồi sức cấp cứu và Chống độc	Thông khí nhân tạo xâm nhập [giờ theo thực tế]	01.0132.0209
4	Hồi sức cấp cứu và Chống độc	Thông khí nhân tạo xâm nhập phương thức PCV [giờ theo thực tế]	01.0133.0209
5	Hồi sức cấp cứu và Chống độc	Thông khí nhân tạo xâm nhập phương thức VCV [giờ theo thực tế]	01.0134.0209
6	Hồi sức cấp cứu và Chống độc	Thông khí nhân tạo xâm nhập phương thức A/C (VCV) [giờ theo thực tế]	01.0135.0209

Figure 11: View Categories list.

- Create Categories:

- In that category page, click to button “Thêm danh mục”.
- Input all of information need.
- Click to “Lưu” button

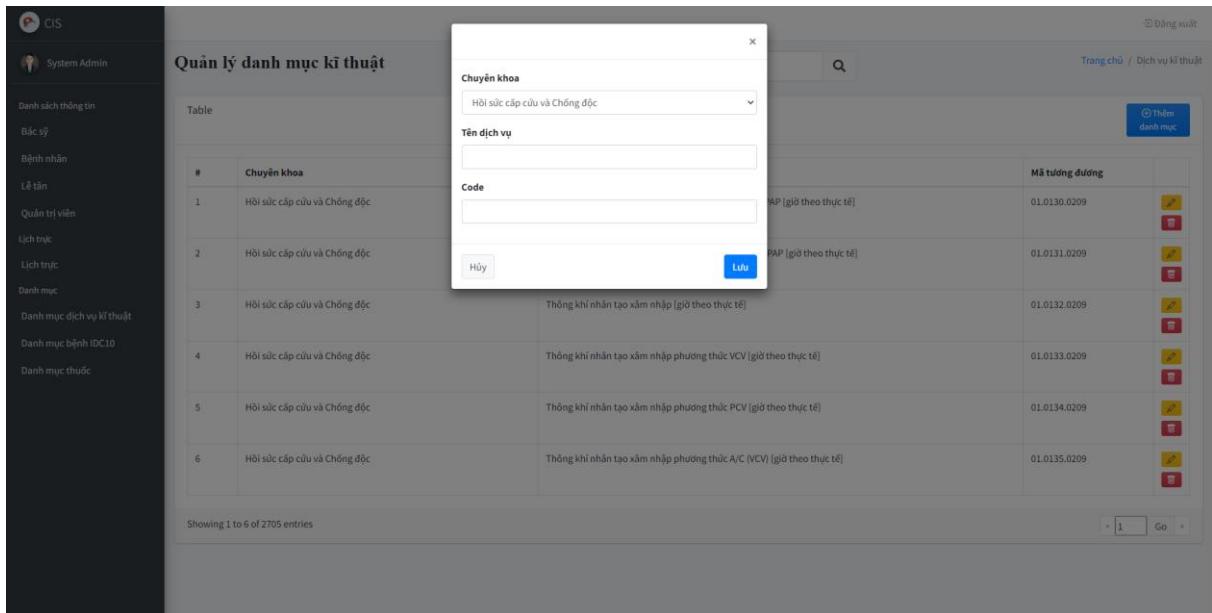


Figure 12: Create Categories.

- Search Categories

- In category page, input keyword in search bar to search Category by title

#	Chuyên khoa	Tên dịch vụ	Mã tương đương
1	Hồi sức cấp cứu và Chống độc	Đặt canuyn mũi hẫu, miệng hẫu	01.0053.0075
2	Nội khoa	Nội soi thực quản - Dạ dày - Tá tràng qua đường mũi	02.0255.0319
3	Nhi khoa	Điều nhí châm điều trị viêm mũi dị ứng	03.0383.0230
4	Nhi khoa	Cấy chỉ điều trị viêm mũi dị ứng	03.0434.0227
5	Nhi khoa	Điều châm điều trị viêm mũi xoang	03.0518.0230
6	Nhi khoa	Thuỷ châm điều trị viêm mũi dị ứng	03.0565.0271

Figure 13: Search Categories.

- Update Categories

- In that category page, click icon edit to go to update Categories page.
- Edit all of information need.
- Click to “Lưu”.

Figure 14: Update Categories

- Delete Categories

- In that category page, click icon delete.
- Click “Có” button when display popup.

Figure 15: Delete Categories.

- View users list

- Admin login.

- Click “Bệnh nhân” to display list of patients.
- Click “Lễ tân” to display list of receptionists.
- Click “Bác sĩ” to display list of doctors.

#	Mã bệnh nhân	Tên	Email	Điện thoại	Ngày sinh	Giới tính	
1	75	Bùi Trí	tribm14@gmail.com	0385617414	02/03/1999	Nam	
2	72	Việt Hoàng	vieth12@gmail.com	0385617412	09/03/2001	Nam	
3	76	Cao Duy	duycao15@gmail.com	0385617415	11/12/1996	Nam	
4	77	Trần Thùy	thuytran16@gmail.com	0385617416	04/06/1986	Nữ	
5	91	Lê Tuấn Anh		0355558474	10/01/1997	Nam	
6	92	Hoàng Linh	hoaqlinh1@gmail.com	0369782088	01/01/1999	Nam	
7	93	Hoàng Việt	viethqhd@gmail.com	0395558546	01/01/1999	Nam	
8	1	System Admin	admin@gmail.com	0123456789	12/01/1999	Nam	
9	96	Bùi Hữu Minh Trí		0352524027	01/01/1999	Nam	

Figure 16: View users list.

- Search users:
 - On page “Bệnh nhân” or “Bác sĩ” or “Lễ tân”.
 - Input keyword in search bar to search users by name.

#	Mã bệnh nhân	Tên	Email	Điện thoại	Ngày sinh	Giới tính	
1	96	Bùi Hữu Minh Trí		0352524027	01/01/1999	Nam	

Figure 17: Search users.

- Create user account:
 - Admin login.
 - On page “Bệnh nhân” or “Bác sĩ” or “Lễ tân”.
 - Click “Thêm người dùng”
 - Input information of user
 - Click “Lưu” button

Điện thoại	Ngày sinh	Giới tính
0323232322	10/10/2002	Nam
0983900442	10/02/1974	Nam
0958456378	11/03/1988	Nam
0356954781	12/03/1977	Nam
0355847531	01/03/1985	Nữ
0759635894	11/03/1956	Nam

Figure 18: Create user account.

- Delete user account:
 - Admin login.
 - Click “Bệnh nhân” or “Bác sĩ” or “Lễ tân”
 - Click icon delete
 - Click “Có” button in popup

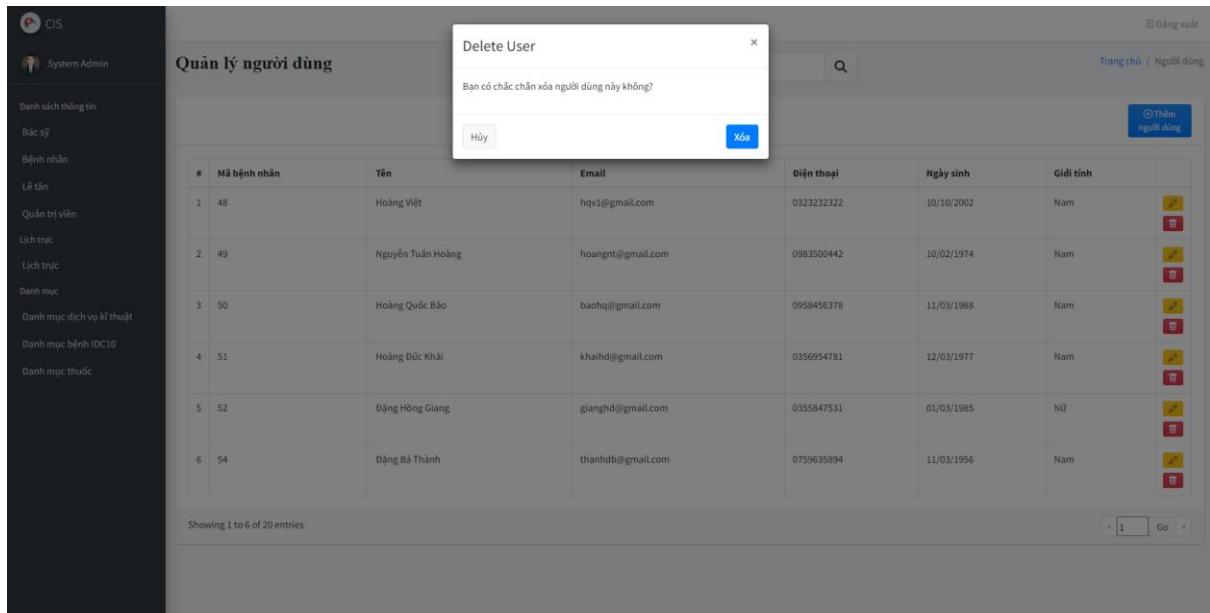


Figure 19: Delete user account.

- View Schedule

- In the left sidebar of website, doctor click “Lịch trực”
- Admin view schedule of doctors.

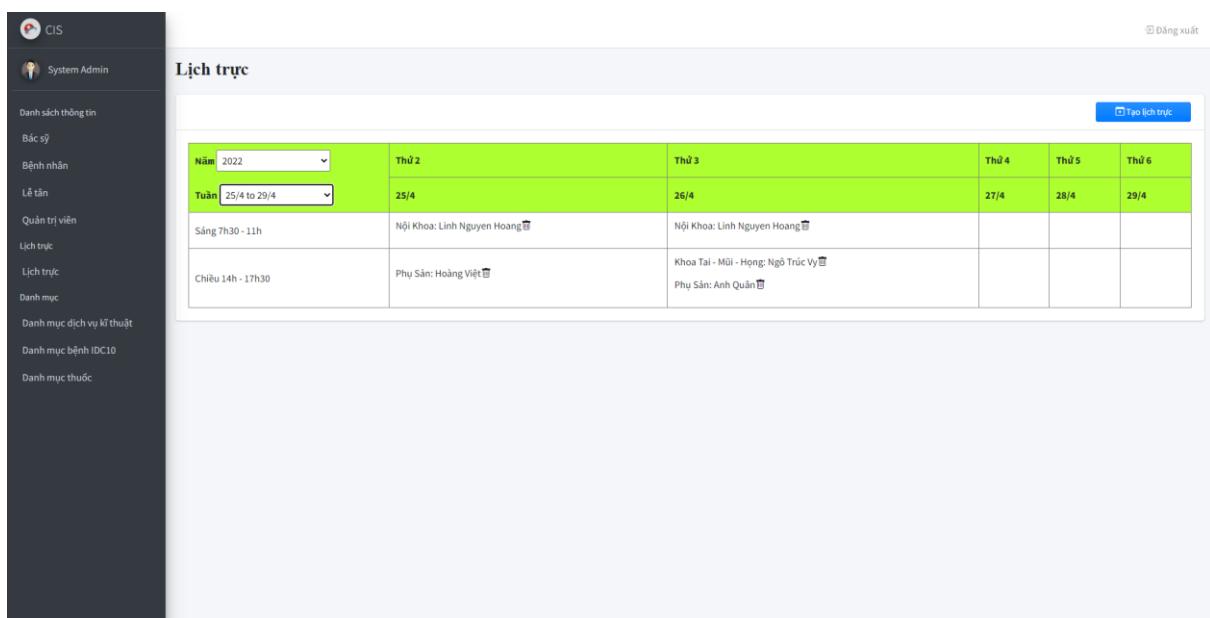


Figure 20: Delete user account.

- Logout:

- In home page

- Click to “Đăng xuất” on the top right of the website.

Lịch trực

Năm	Thứ 2	Thứ 3	Thứ 4	Thứ 5	Thứ 6
2022	25/4	26/4	27/4	28/4	29/4
Tuần	Sáng 7h30 - 11h Chiều 14h - 17h30	Nội Khoa: Linh Nguyen Hoang Phụ Sản: Hoàng Việt	Nội Khoa: Linh Nguyen Hoang Khoa Tai - Mũi - Họng: Ngô Trúc Vy Phụ Sản: Anh Quân		

Figure 21: Logout screen

3.2.5 Receptionist

- View appointment list:
 - Receptionist login
 - Click to “Danh sách lịch hẹn” in the left sidebar of website
 - Receptionist view appointment list with their informations.

Lịch hẹn

Mã đặt lịch	Số điện thoại	Họ và tên	Thời gian đặt lịch	Trạng thái
40	0698456214	Nguyễn Văn Kiên	25/04/2022	Đã hủy
97	0388888881	Nguyễn Hoàng Linh	25/04/2022	Đã hủy
193	0389999999	Hoàng Vũ	25/04/2022	Đã hủy
334	0388888881	Nguyễn Hoàng Anh Tài	25/04/2022	Đang chờ
341	0904380878	Trần Thu Thủy	25/04/2022	Đã xác nhận
343	0904380878	Trần Thu Thủy	25/04/2022	Đang chờ

Figure 22: View appointment list

- Make a new appointment:
 - On Patients list screen
 - Click to “Tạo lịch hẹn” in the right top of website
 - Enter all informations of appointment.
 - Click to “Xác nhận lịch hẹn” to make a new appointment.

ĐẶT LỊCH KHÁM

Số điện thoại (*)

Giới tính (*)

Họ và tên (*)

Email

Ngày sinh (*)

Tên người Giám hộ (nếu có)

Địa chỉ

Số điện thoại Giám hộ (nếu có)

Chọn ngày khám (*)

Mối quan hệ với người bệnh (nếu có)

Chọn giờ khám (*)

Văn đề sức khỏe cần khám

Chọn phòng

Trạng thái:

Xác nhận lịch hẹn Tiếp nhận Quay lại

Figure 23: Make a new appointment

- View patients list:
 - Receptionist login
 - Click to “Danh sách bệnh nhân” in the left sidebar of website
 - Receptionist view patients list with their informations.

Quản lý người dùng

Type your keywords here

Thêm người dùng

#	Mã bệnh nhân	Tên	Email	Điện thoại	Ngày sinh	Giới tính
1	75	Bùi Trí	tribm14@gmail.com	0385617414	02/03/1999	Nam
2	72	Việt Hoàng	vieth12@gmail.com	0385617412	09/03/2001	Nam
3	76	Cao Duy	duycao15@gmail.com	0385617415	11/12/1996	Nam
4	77	Trần Thúy	thuytran16@gmail.com	0385617416	04/06/1986	Nữ
5	91	Lê Tuấn Anh		035558474	10/01/1997	Nam
6	92	Hoàng Linh	hoaqlinh1@gmail.com	0369782088	01/01/1999	Nam

Showing 1 to 6 of 41 entries

Figure 24: View patients list

- Payment:
 - Receptionist login
 - Click to “Thanh toán” in the left sidebar of website
 - Receptionist view patients list with their bill detail.

Thanh toán

HÓA ĐƠN VIỆN PHÍ

Họ và tên: Bùi Hữu Minh Trí Tuổi: 23 Giới tính: Nam

Địa chỉ: BN

Phòng Khám

#	Tên dịch vụ	Số lượng	Đơn giá	Thành tiền
1	Tim tế bào Hargraves	1	64000	64000
2	Bơm rửa khoang màng phổi	1	64000	64000
3	Chọc tháo dịch màng phổi dưới hướng dẫn của siêu âm	1	64000	64000

Tổng chi phí (VND) 192000

Người nộp (Ký, ghi rõ họ tên)

... Ngày ... tháng ... năm ...

Thư Ngan (Ký, ghi rõ họ tên)

Hủy Thanh toán In hóa đơn

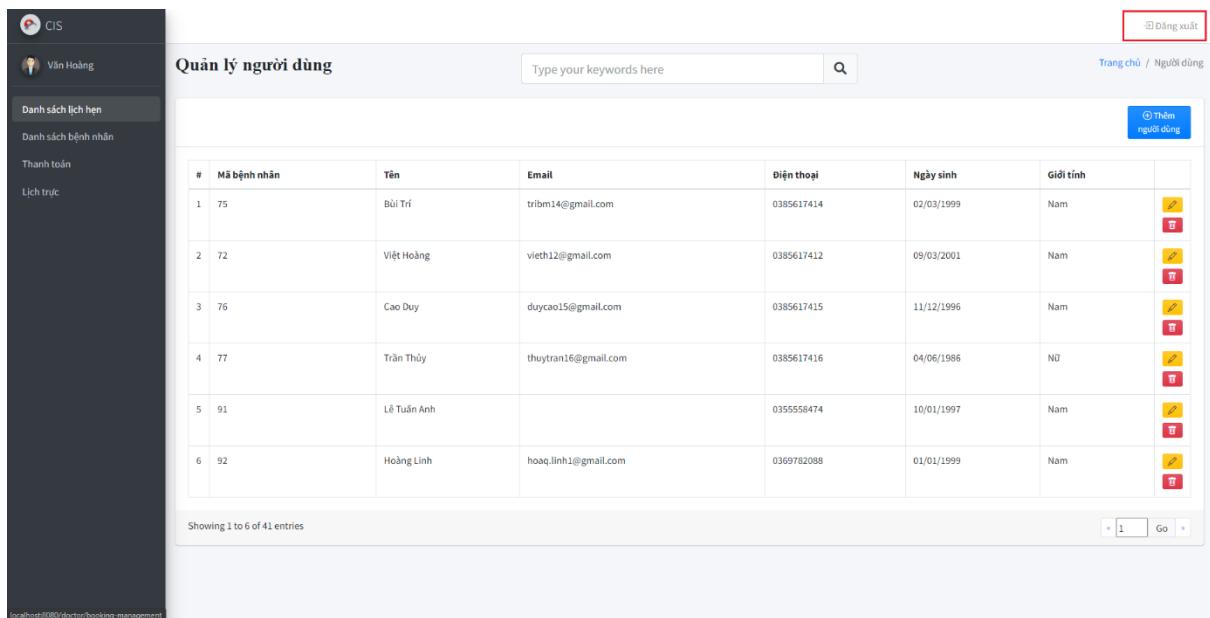
Đăng xuất

Trang chủ / Thanh toán

Ngày khám	Trạng thái	Chi tiết
05/2023	Thanh toán	
04/2022	Chưa thanh toán	
12/2022	Chưa thanh toán	
04/2022	Thanh toán	
01/2022	Chưa thanh toán	
04/2022	Chưa thanh toán	

Figure 25: View payment & print bill

- Sign out
 - In the page, Receptionist chooses the “Đăng xuất”.



The screenshot shows a user management interface. At the top right, there is a red box around the "Đăng xuất" (Logout) button. The main area displays a table of users with columns: #, Mã bệnh nhân (Patient ID), Tên (Name), Email, Điện thoại (Phone), Ngày sinh (Birth Date), Giới tính (Gender), and two small icons. The table contains 6 entries. At the bottom left, it says "Showing 1 to 6 of 41 entries". On the far left, there's a sidebar with links like "Danh sách lịch hẹn", "Danh sách bệnh nhân", "Thanh toán", and "Lịch trực". The URL at the bottom is "localhost:8080/doctor/booking-management".

#	Mã bệnh nhân	Tên	Email	Điện thoại	Ngày sinh	Giới tính	
1	75	Bùi Trí	tribm14@gmail.com	0385617414	02/03/1999	Nam	
2	72	Việt Hoàng	vieth12@gmail.com	0385617412	09/03/2001	Nam	
3	76	Cao Duy	duycao15@gmail.com	0385617415	11/12/1996	Nam	
4	77	Trần Thúy	thuytran16@gmail.com	0385617416	04/06/1986	Nữ	
5	91	Lê Tuấn Anh		0355558474	10/01/1997	Nam	
6	92	Hoàng Linh	hoaqlinh1@gmail.com	0369782088	01/01/1999	Nam	

Figure 26: Sign out

3.3 Troubleshooting

The application can be slow when the internet is poor.

VII. Appendix

Reference:

FMP website: <https://www.vietnammedicalpractice.com/>

ISOHCARE: <https://isofhcicare.com/>